Title: Deep Predictive Learning as a Model of Human Learning

Authors: Randall C. O'Reilly,* Jacob L. Russin, and John Rohrlich

Affiliations: Departments of Psychology and Computer Science
Center for Neuroscience
University of California, Davis

*Correspondence to: oreilly@ucdavis.edu
1544 Newton Ct
Davis, CA 95616

September 26, 2019

Abstract: Understanding the principles underlying the power of human learning is a widely held goal of research in machine learning, yet these models unrealistically rely on massive human-labeled datasets. We present a biologically based model of predictive learning, which generates predictions at the alpha frequency (100 msec), and learns from prediction errors, requiring no labeled inputs. Distinctive patterns of connectivity between the neocortex and thalamus drive alternating top-down prediction and bottom-up outcome representations over the pulvinar nucleus, with the temporal difference driving error-driven learning throughout neocortex. However, can it learn abstractions that go beyond the surface input level? Our model does, in ways that match both human category representations and monkey electrophysiology, while comparison models lacking biological features do not.

One Sentence Summary: The human brain learns from prediction errors generated on the pulvinar nucleus, and a model thereof develops abstract knowledge.

The success of deep convolutional neural networks (DCNN's) (*1–3*) in object recognition and many other domains raises the question of how well they model human learning, at both neural and cognitive levels. Although the engine of these models, error backpropagation (*4*), has long been questioned on biological grounds (*5*), various related biologically plausible mechanisms have been proposed (*6–8*). However, the need for massive amounts of labeled data still makes these models cognitively implausible: non-human primates and infants learn to recognize and categorize objects without the benefit of such labeled data (*9*). An alternative, biologically plausible approach is to use *predictive error-driven learning*, where error signals arise from differences between a prediction of what will happen next, and what actually does occur (*10, 11*). In principle, all that this requires is: 1) for events to unfold over time; 2) a learning system that is somehow organized to generate predictions of these events; and 3) a biological mechanism that learns from prediction errors. Furthermore, a system that learned to accurately predict complex real-world events would require considerable knowledge to have been acquired in the process of so doing, and thus there is reason to believe that predictive learning could power sophisticated, important forms of developmental learning.

Here we show that canonical circuits between the neocortex and thalamus have several distinctive properties that directly support predictive error-driven learning. When implemented in a computational model employing a biologically plausible form of error backpropagation (*6, 12, 13*), along with several other important properties of the mammalian visual system, the model learns to systematically categorize 3D objects according to invariant shape properties. Furthermore, this category structure matches human judgments of these same objects, and is consistent with neural representations in inferotemporal (IT) cortex in primates. Comparison models with the same architecture but using standard non-biological error-backpropagation learning, and models using the state-of-the-art *PredNet* predictive learning architecture (*14*), support the idea that predictive learning is useful for shaping internal representations, but these models do not learn much beyond the similarities present at the lowest visual levels. Thus, we argue that incorporating biological properties of the brain can potentially provide a better understanding of human learning at multiple levels relative to existing DCCN models.

Motivated by biological evidence, we hypothesize that sensory predictions in posterior neocortex are generated roughly every 100 msec (i.e., the *alpha* rhythm, 10 Hz), by neurons in the deep layers of the neocortex that project to the pulvinar nucleus of the thalamus (Figure 1a) (*15*). The pulvinar represents this top-down prediction for roughly 75 msec of the alpha cycle as it develops, after which point the layer 5IB intrinsic-bursting neurons send strong, bottom-up driving input to the pulvinar, representing the actual sensory stimulus (*16*). These 5IB neurons burst at the alpha frequency, determining the overall timing of the predictive learning cycle, along with other dynamic parameters of the thalamocortical circuit (*17–19*). The prediction error is implicit in the temporal difference between these two periods of activity within the alpha cycle over the pulvinar, which is consistent with the biologically plausible form of error-driven cortical learning used in our models (*6*). The pulvinar sends broad projections back up to all of the areas that drive top-down predictions into it (*20, 21*), thus broadcasting this error signal to drive local synaptic plasticity in the neocortex. This mathematically approximates gradient descent to minimize overall prediction errors. This computational framework makes sense of otherwise puzzling anatomical and physiological properties of the cortical and thalamic networks (*16*), and is consistent with a wide range of detailed neural and behavioral data regarding the effects of the alpha rhythm on learning and

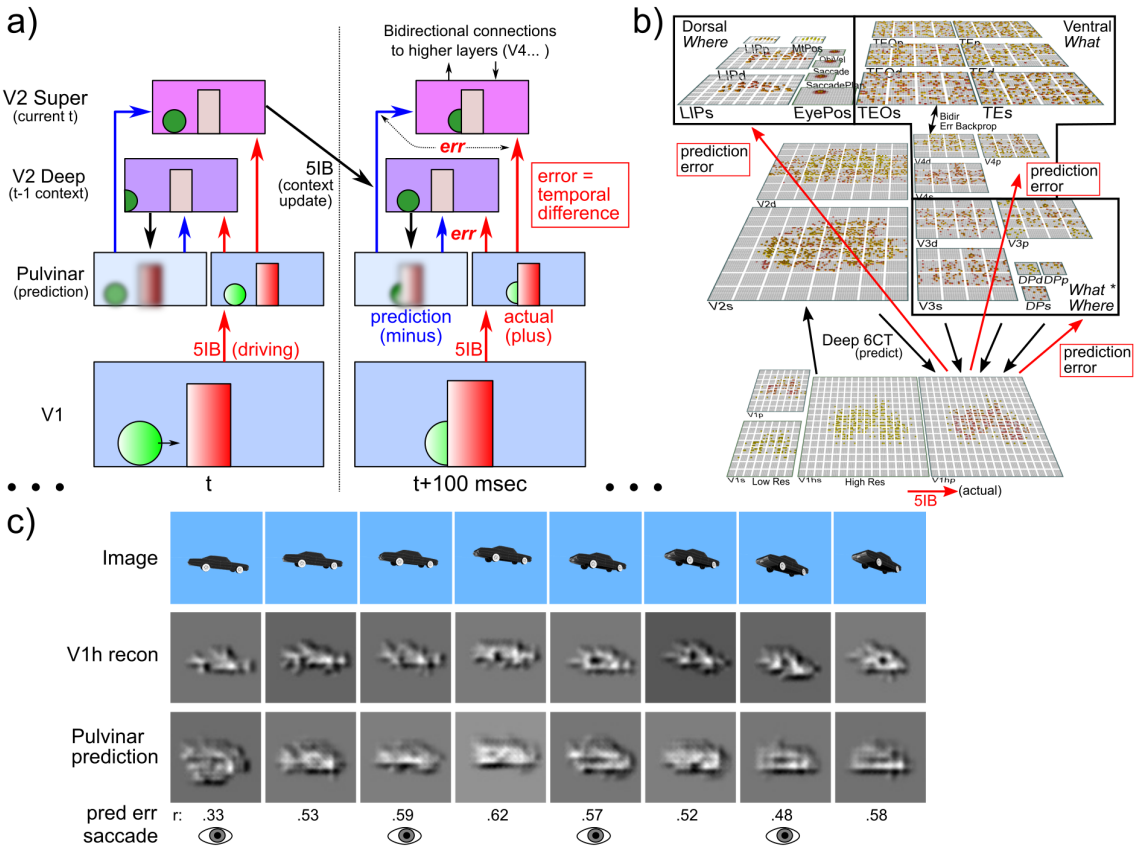

Figure 1: **a)** Schematic illustration of the temporal evolution of information flow in the DeepLeabra model predicting visual sequences, over two alpha cycles of 100 msec each. During each alpha cycle, the V2 Deep layer (cortical lamina 5, 6) uses the prior 100 msec of context information to generate a prediction or expectation (minus phase) over the pulvinar thalamic relay cell (TRC) neurons, of what will happen next. The bottom-up actual outcome is driven over the pulvinar via the 5IB strong driver inputs from V1, providing the *plus* or *target* phase of learning. Error-driven learning occurs as a function of the *temporal difference* between the plus and minus activation states, in both superficial (lamina 2, 3) and deep layers, via the pulvinar projections into these layers. The 5IB bursting in V2 drives an update of the local temporal context information in V2 Deep layers, which is used in generating the minus phase prediction in the next alpha cycle, and so on. These same 5IB cells drive a plus phase in higher area TRC's as well, which perform the same kind of *local* predictive auto-encoder learning as shown for V2 here. See supplementary material for more details. **b)** The three-visual-stream deep predictive learning model (*What-Where-Integration, WWI* model). The dorsal *Where* pathway learns first, using easily-abstracted *spatial blobs*, to predict where an object will move next, based on prior motion history, visual motion, and saccade efferent copy signals. This drives strong top-down inputs to lower areas with accurate spatial predictions, leaving the *residual* error concentrated on *What* and *What * Where* integration information. The V3 and DP (dorsal prelunate) areas constitute the *What * Where* integration pathway, helping bind features and locations. V4, TEO, and TE are the *What* pathway, learning abstracted object category representations, which also drive strong top-down inputs to lower areas. *s* suffix = superficial layer, *d* = deep layer, and *p* = pulvinar. **c)** An example sequence of 8 frames (8 alpha cycles) that the model learned to predict, with the reconstruction of each image based on the V1 gabor filters (*V1 recon*), and a reconstruction of the model-generated prediction for each frame over the higher resolution V1hp pulvinar layer (to compare against V1 recon, correlation value r shown). The relatively low resolution encoding of the image makes these somewhat difficult to interpret, but the r values are well above the r's for each V1 state compared to the previous time step (mean = .38, min of .16 on frame 4 when the prediction is at .57 – see supplementary material for more analysis), indicating that the model has learned somewhat vague but broadly accurate predictions that go beyond e.g., just copying the previous time step. The eye icons indicate when a saccade occurred.

perception (*22–25*). It has many testable differences from other existing theories of predictive learning that have been proposed over the years, at varying levels of biological detail (*26–29*).

A critical question for predictive learning is whether it can develop high-level, abstract ways of representing the raw sensory inputs, while learning from nothing but predicting these low-level visual inputs. For example, can predictive learning really eliminate the need for human-labeled image datasets where abstract category information is explicitly used to train object recognition models via error-backpropagation? From a cognitive perspective, there is considerable evidence that non-verbal primates, and pre-verbal human infants, naturally develop abstract categorical encodings of visual objects in IT cortex (*30*), without relying on any explicit external categorical labels. Existing predictive-learning models based on error backpropagation (*14*) have not demonstrated the development of abstract, categorical representations. Previous work has shown that predictive learning can be a useful method for pretraining networks that are subsequently trained using human-generated labels, but here we focus on the formation of systematic categories *de-novo*.

To determine if our biologically based predictive learning model (Figure 1b) can naturally form such categorical encodings in the complete absence of external category labels, we showed the model brief movies of 156 3D object exemplars drawn from 20 different basic-level categories (e.g., car, stapler, table lamp, traffic cone, etc.) selected from the CU3D-100 dataset (*31*). The objects moved and rotated in 3D space over 8 movie frames, where each frame was sampled at the alpha frequency (Figure 1c). There were also saccadic eye movements every other frame, with an efferent copy signal to enable full prediction of the effects of the eye movement, which allows the model to capture predictive remapping (a widely-studied signature of predictive learning in the brain) (*32, 33*), and introduces additional predictive-learning challenge. The only learning signal available to the model was the temporal difference prediction error between what it predicted to see in the next frame, compared to what was actually seen.

We performed a representational similarity analysis (RSA) on the learned activity patterns at each layer in the model, and found that the highest IT layer (TE) produced a systematic organization of the 156 3D objects into 5 categories (Figure 2a), which visually correspond to the overall shape of the objects (pyramid-shaped, vertically-elongated, round, boxy / square, and horizontally-elongated). This organization of the objects matches that produced by humans making shape similarity judgments on the same set of objects, using the V1 reconstruction as shown in Figure 1c to capture the model's coarse-grained perception (Figure 2b; see supplementary material for methods and further analysis). Critically, Figure 2c shows that the overall similarity structure present in IT layers (TEO, TE) of the biological model is significantly different from the similarity structure at the level of the V1 primary visual input. Thus the model, despite being trained only to generate accurate visual input-level predictions, has learned to represent these objects in an abstract way that goes beyond the raw input-level information. Furthermore, because this abstract category organization reflects the overall visual shapes of the objects as judged by human participants, this suggests that the model is extracting the geometrical shape information that is apparent once these objects are encoded in representations that are invariant to differences in motion, rotation, and scaling present in the V1 visual inputs. We further verified that at the highest IT levels in the model, a consistent, spatially-invariant representation is present across different views of the same object (e.g., the average correlation across frames within an object was .901). This is also evident in Figure 2a by virtue of the close similarity across multiple objects within the same category.

Figure 2: **a)** Category similarity structure that developed in the highest layer, TE, of the biologically based predictive learning model, showing *1-correlation* similarity of the TE representation for each 3D object against every other 3D object (156 total objects). Blue cells have high similarity, and model has learned block-diagonal clusters or categories of high-similarity groupings, contrasted against dissimilar off-diagonal other categories. Clustering maximized the overall average *within - between* correlation distance across given set of clusters (see supplementary materials for details). Note that all items from the same "objective" basic-level object categories (N=20) are reliably sorted within a given learned category, so these categories subsume our more fine-grained object categories. **b)** Human similarity ratings for the same 3D objects, presented with the V1 reconstruction (see Fig 1c) to capture coarse perception in our model, aggregated at the level of the 20 basic-level categories since we could not sample the entire 156 x 156 matrix. Each cell is 1 - proportion of time the given pair of objects was rated as more similar than another pair of objects (see supplementary material for details of the experiment). The resulting similarity matrix generally exhibits the same categorical structure as the model (confirmed by permutation testing and agglomorative cluster analysis). **c)** Emergence of abstract category structure over the hierarchy of layers. Red line shows correlation similarity between the similarity matrix for TE (shown in panel a) against the similarity matrix computed for every other layer, and the black line shows the correlation similarity for the V1 layer matrix against every other layer (1 = identical; 0 = orthogonal). Both show that IT layers (TEO, TE) progressively differentiate from raw input similarity structure present in V1, and, critically, that the model has learned structure beyond that present in the input.

Further evidence for the progressive nature of representation development in our model is shown in Figure 3, which compares the similarity structures in layers V4 and IT in macaque monkeys (*30*) with those in corresponding layers in our model. In both the monkeys and our model, the higher IT layer builds upon and clarifies the noisier structure that is emerging in the earlier V4 layer. Considerable other work has also compared DCNN representations with these same data from monkeys (*30*), but it is essential to appreciate that those DCNN models were explicitly trained on the category labels, making it somewhat less than surprising that such categorical representations developed. By contrast, we reiterate that our model has discovered its categorical representations entirely on its own, with no explicit categorical inputs or training of any kind.

Figure 4 shows the results from a purely backpropagation-based version of the same model architecture. In this model, the highest layers in the network form a simple binary category structure overall, and the detailed item-level similarity structure does not diverge significantly from that present at the lowest V1 inputs, indicating that it has not formed novel systematic structured representations, in contrast to those formed in the biologically based model. Thus, it is clear that the

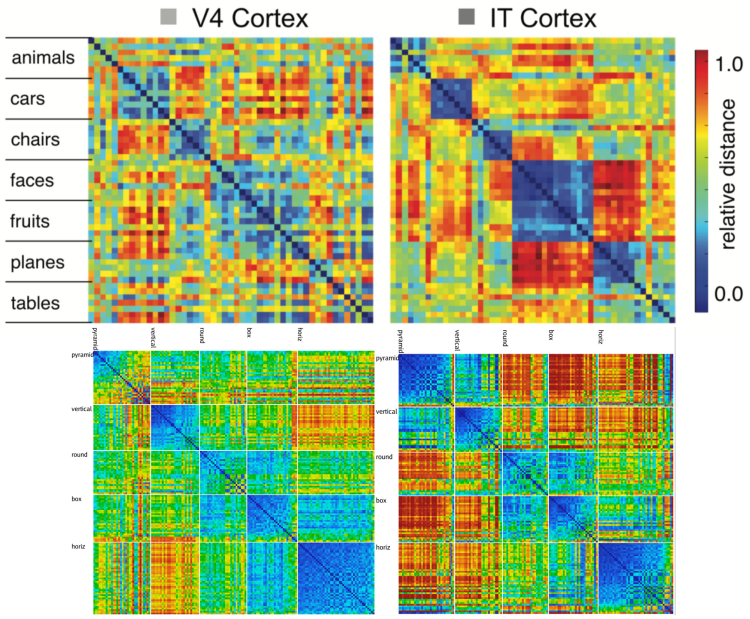

Figure 3: Comparison of progression from V4 to IT in macaque monkey visual cortex (top row, from Cadieu et al, 2014) versus same progression in model (replotted using comparable color scale). Although the underlying categories are different, and the monkeys have a much richer multi-modal experience of the world that could help reinforce categories such as foods and faces, our model nevertheless shows a similar qualitative progression in extent of stronger categorical structure in IT, where the block-diagonal highly similar representations are more consistent across categories, and the off-diagonal differences are stronger and more consistent as well (i.e., categories are also more clearly differentiated). Note that the critical difference in our model versus those compared in Cadieu et al 2014 and related papers is that they explicitly trained their models on category labels, whereas our model is *entirely self-organizing* and has no external categorical training signal.

additional biologically motivated properties of the original model are playing a critical role in the development of abstract categorical representations. These properties include: excitatory bidirectional connections, inhibitory competition, and an additional Hebbian form of learning that serves as a regularizer (similar to weight decay) on top of predictive error-driven learning (*12, 34*).

Each of these properties could promote the formation of categorical representations. Bidirectional connections enable top-down signals to consistently shape lower-level representations, creating significant attractor dynamics that cause the entire network to settle into discrete categorical attractor states. By contrast, backpropagation networks typically lack these kinds of attractor dynamics, and this could contribute significantly to their relative lack of categorical learning. Hebbian learning drives the formation of representations that encode the principal components of activity correlations over time, which can help more categorical representations coalesce (and results below already indicate its importance). Inhibition, especially in combination with Hebbian learning, drives representations to specialize on more specific subsets of the space. Ongoing work is attempting to determine which of these is essential in this case (perhaps all of them) by systematically introducing some of these properties into the backpropagation model, though this is difficult because full bidirectional recurrent activity propagation, which is essential for conveying error signals top-down in the biological network, is incompatible with the standard efficient form of error

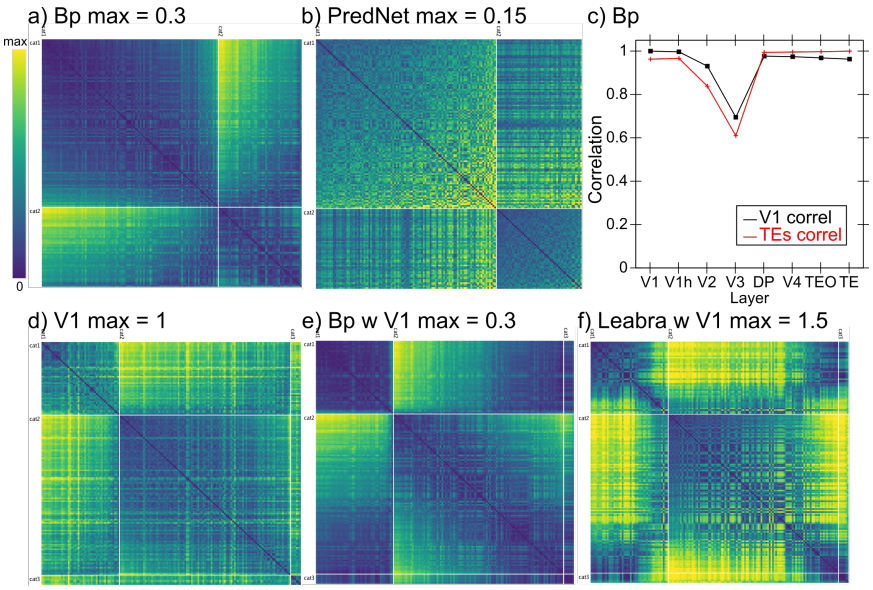

Figure 4: **a)** Category similarity structure in the highest IT layer (TE) of the backpropagation (Bp) model with the same What / Where structure. Only two broad categories are present, and the lower *max* distance (0.3 vs. 1.5 in biological model) indicates that overall the patterns are highly similar. **b)** Similarity structure for the PredNet model, in the highest of its layers (layer 3), which is even less differentiated (max = 0.15) but overall follows the same broad category structure. **c)** Comparison of similarity structures across layers in the Bp model (compare to Figure 2c): unlike in the biological model, the V1 structure is largely preserved across layers, and is little different from the structure that best fits the TE layer shown in panel a, indicating that the model has not developed abstractions beyond the structure present in the visual input. Layer V3 is most directly influenced by spatial prediction errors in its connections with the dorsal pathway, so it differs from both in strongly encoding position information. **d)** The best fitting V1 structure, which has 2 broad categories and banana is in a third category by itself. The lack of dark blue on the block diagonal indicates that these categories are overall quite weak, and every item is fairly dissimilar from every other. **e)** The same similarities shown in panel a for Bp TE also fit reasonably well in the V1 structure (and they have a similar average within - between contrast differneces, of -0.0838 and -0.0513). **f)** The similarity structure from the biological model does *not* fit well within the V1 organization (the blue is not aligned along the block diagonal, and the yellow is not strictly off-diagonal), consistent with the large difference in average contrast distance (0.5071 for the best categories vs. 0.3070 for the V1 categories).

backpropagation, and requires much more computationally intensive and unstable forms of fully recurrent backpropagation (*35, 36*). Furthermore, Hebbian learning requires inhibitory competition which is difficult to incorporate within the backpropagation framework.

Figure 5 shows just a few of the large number of parameter manipulations that have been conducted to develop and test the final architecture. For example, we hypothesized that separating the overall prediction problem between a spatial *Where* vs. non-spatial *What* pathway (*37, 38*), would strongly benefit the formation of more abstract, categorical object representations in the *What* pathway. Specifically, the *Where* pathway can learn relatively quickly to predict the overall spatial trajectory of the object (and anticipate the effects of saccades), and thus effectively regress out that component of the overall prediction error, leaving the residual error concentrated in object feature information, which can train the ventral *What* pathway to develop abstract visual categories.

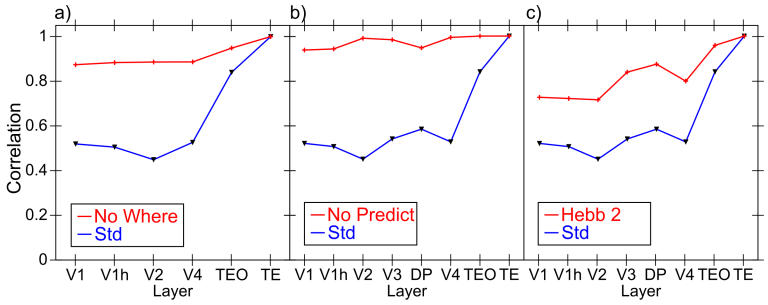

Figure 5: Effects of various manipulations on the extent to which the TE layer representations differentiate from the V1 structure. *Std* is the same result shown in Figure 2c from the intact model (see caption there for further explanation), and all manipulations significantly impair the development of abstract TE categorical representations (i.e., the TE representations are more similar to V1 and the other layers). **a)** Dorsal *Where* pathway lesions, including lateral inferior parietal sulcus (LIP), V3, and dorsal prelunate (DP). This pathway is essential for regressing out location-based prediction errors, so that the residual errors concentrate feature-encoding errors that train the *What* pathway. **b)** Allowing the deep layers full access to current-time information, thus effectively eliminating the prediction demand and turning the network into an auto-encoder, which significantly impairs representation development, and supports the importance of the challenge of predictive learning for developing deeper, more abstract representations. **c)** Reducing the strength of Hebbian learning by 20% (from 2.5 to 2), demonstrating the essential role played by this form of learning on shaping categorical representations. Eliminating Hebbian learning entirely prevented the model from learning anything at all, as it also plays a critical regularization and shaping role on learning.

Figure 5a shows that, indeed, when the *Where* pathway is lesioned, the formation of abstract categorical representations in the intact *What* pathway is significantly impaired. Figure 5b shows that full predictive learning, as compared to just encoding and decoding the current state (which is much easier computationally, and leads to much better overall accuracy), is also critical for the formation of abstract categorical representations — prediction is a "desirable difficulty" (*39*). Finally, Figure 5c shows the impact of reducing Hebbian learning, which impairs category learning as expected.

In conclusion, we have demonstrated that learning based strictly on predicting what will be seen next is, in conjunction with a number of critical biologically motivated network properties and mechanisms, capable of generating abstract, invariant categorical representations of the overall shapes of objects. The nature of these shape representations closely matches human shape similarity judgments on the same objects. Thus, predictive learning has the potential to go beyond the surface structure of its inputs, and develop systematic, abstract encodings of the "deeper" structure of the environment. Relative to existing machine-learning-based approaches in "deep learning", which have generally focused on raw categorization accuracy measures using explicit category labels or other human-labeled inputs, the results here suggest that focusing more on the nature of what is learned in the model might provide a valuable alternative approach. Considerable evidence in cognitive neuroscience suggests that the primary function of the many nested ("deep") layers of neural processing in the neocortex is to *simplify* and aggressively *discard* information (*40*), to produce precisely the kinds of extremely valuable abstractions such as object categories, and, ultimately, symbol-like representations that support high-level cogntive processeses such as reasoning and problem-solving (*41, 42*). Thus, particularly in the domain of predictive or generative learn-

ing, the metric of interest should not be the accuracy of prediction itself (which is indeed notably worse in our biologically based model compared to the DCNN-based PredNet and backpropagation models), but rather whether this learning process results in the formation of simpler, abstract representations of the world that can in turn support higher levels of cognitive function.

Considerable further work remains to be done to more precisely characterize the essential properties of our biologically motivated model necessary to produce this abstract form of learning, and to further explore the full scope of predictive learning across different domains. We strongly suspect that extensive cross-modal predictive learning in real-world environments, including between sensory and motor systems, is a significant factor in infant development and could greatly multiply the opportunities for the formation of higher-order abstract representations that more compactly and systematically capture the structure of the world (*43*). Future versions of these models could thus potentially provide novel insights into the fundamental question of how deep an understanding a pre-verbal human, or a non-verbal primate, can develop (*11, 44*), based on predictive learning mechanisms. This would then represent the foundation upon which language and cultural learning builds, to shape the full extent of human intelligence.

References

1. A. Krizhevsky, I. Sutskever, G. E. Hinton, *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger, eds. (Curran Associates, Inc., 2012), pp. 1097–1105.
2. Y. LeCun, Y. Bengio, G. Hinton, *Nature* **521**, 436 (2015).
3. J. Schmidhuber, *Neural Networks* **61**, 85 (2015).
4. D. E. Rumelhart, G. E. Hinton, R. J. Williams, *Nature* **323**, 533 (1986).
5. F. Crick, *Nature* **337**, 129 (1989).
6. R. C. O'Reilly, *Neural Computation* **8**, 895 (1996).
7. X. Xie, H. S. Seung, *Neural Computation* **15**, 441 (2003).
8. Y. Bengio, T. Mesnard, A. Fischer, S. Zhang, Y. Wu, *Neural Computation* **29**, 555 (2017).
9. B. M. Lake, T. D. Ullman, J. B. Tenenbaum, S. J. Gershman, *Behavioral and Brain Sciences* **40** (2017/ed).
10. J. L. Elman, *Cognitive Science* **14**, 179 (1990).
11. J. Elman, *et al.*, *Rethinking Innateness: A Connectionist Perspective on Development* (MIT Press, Cambridge, MA, 1996).
12. R. C. O'Reilly, Y. Munakata, *Computational Explorations in Cognitive Neuroscience: Understanding the Mind by Simulating the Brain* (MIT Press, Cambridge, MA, 2000).

13. R. C. O'Reilly, Y. Munakata, M. J. Frank, T. E. Hazy, Contributors, *Computational Cognitive Neuroscience* (Wiki Book, 1st Edition, URL: http://ccnbook.colorado.edu, 2012).

14. W. Lotter, G. Kreiman, D. Cox, *arXiv:1605.08104 [cs, q-bio]* (2016).

15. R. C. O'Reilly, D. Wyatte, J. Rohrlich, *arXiv:1407.3432 [q-bio]* (2014).

16. S. Sherman, R. Guillery, *Exploring the Thalamus and Its Role in Cortical Function* (MIT Press, Cambridge, MA, 2006).

17. M. L. Lorincz, K. A. Kekesi, G. Juhasz, V. Crunelli, S. W. Hughes, *Neuron* **63**, 683 (2009).

18. S. Franceschetti, *et al.*, *Brain Research* **696**, 127 (1995).

19. Y. B. Saalmann, M. A. Pinsk, L. Wang, X. Li, S. Kastner, *Science* **337**, 753 (2012).

20. S. Shipp, *Philosophical Transactions of the Royal Society of London B* **358**, 1605 (2003).

21. D. Mumford, *Biological Cybernetics* **65**, 135 (1991).

22. E. A. Buffalo, P. Fries, R. Landman, T. J. Buschman, R. Desimone, *Proceedings of the National Academy of Sciences of the United States of America* **108**, 11262 (2011).

23. R. VanRullen, C. Koch, *Trends in Cognitive Sciences* **7**, 207 (2003).

24. O. Jensen, M. Bonnefond, R. VanRullen, *Trends in Cognitive Sciences* **16**, 200 (2012).

25. I. C. Fiebelkorn, S. Kastner, *Trends in Cognitive Sciences* **23**, 87 (2019).

26. D. Mumford, *Biological Cybernetics* **66**, 241 (1992).

27. R. P. Rao, D. H. Ballard, *Nature Neuroscience* **2**, 79 (1999).

28. M. Kawato, H. Hayakawa, T. Inui, *Network: Computation in Neural Systems* **4**, 415 (1993).

29. K. Friston, *Philosophical Transactions of the Royal Society B* **360**, 815 (2005).

30. C. F. Cadieu, *et al.*, *PLoS Computational Biology* **10**, e1003963 (2014).

31. R. C. O'Reilly, D. Wyatte, S. Herd, B. Mingus, D. J. Jilk, *Frontiers in Psychology* **4** (2013).

32. J. R. Duhamel, C. L. Colby, M. E. Goldberg, *Science* **255**, 90 (1992).

33. P. Cavanagh, A. R. Hunt, A. Afraz, M. Rolfs, *Trends in Cognitive Sciences* **14**, 147 (2010).

34. R. C. O'Reilly, *Trends in Cognitive Sciences* **2**, 455 (1998).

35. R. J. Williams, D. Zipser, *Backpropagation: Theory, Architectures and Applications*, Y. Chauvin, D. E. Rumelhart, eds. (Erlbaum, Hillsdale, NJ, 1992).

36. F. J. Pineda, *Physical Review Letters* **18**, 2229 (1987).

37. L. G. Ungerleider, M. Mishkin, *The Analysis of Visual Behavior*, D. J. Ingle, M. A. Goodale, R. J. W. Mansfield, eds. (MIT Press, Cambridge, MA, 1982), pp. 549–586.

38. M. A. Goodale, A. D. Milner, *Trends in Neurosciences* **15**, 20 (1992).

39. R. A. Bjork, *Metacognition: Knowing about Knowing* (The MIT Press, Cambridge, MA, US, 1994), pp. 185–205.

40. D. J. Simons, R. A. Rensink, *Trends in cognitive sciences* **9**, 16 (2005).

41. N. P. Rougier, D. Noelle, T. S. Braver, J. D. Cohen, R. C. O'Reilly, *Proceedings of the National Academy of Sciences* **102**, 7338 (2005).

42. R. C. O'Reilly, *et al.*, *The Architecture of Cognition: Rethinking Fodor and Pylyshyn*[1]*s Systematicity Challenge*, I. P. Calvo, J. Symons, eds. (MIT Press, Cambridge, MA, 2014).

43. C. Yu, L. B. Smith, *Cognition* **125**, 244 (2012).

44. E. Spelke, K. Breinlinger, J. Macomber, K. Jacobson, *Psychological Review* **99**, 605 (1992).

Acknowledgements

We thank Dean Wyatte, Tom Hazy, Seth Herd, Kai Krueger, Tim Curran, David Sheinberg, Lew Harvey, Jessica Mollick, Will Chapman, Helene Devillez, and the rest of the CCN Lab for many helpful comments and suggestions.

Funding: Supported by: ONR grants ONR N00014-19-1-2684 / N00014-18-1-2116, N00014-14-1-0670 / N00014-16-1-2128, N00014-18-C-2067, N00014-13-1-0067, D00014-12-C-0638. This work utilized the Janus supercomputer, which is supported by the National Science Foundation (award number CNS-0821794) and the University of Colorado Boulder. The Janus supercomputer is a joint effort of the University of Colorado Boulder, the University of Colorado Denver and the National Center for Atmospheric Research.

Author Contributions: RCO developed the model, performed the non-PredNet simulations, and drafted the paper. JLW performed the PredNet simulations and analysis, and edited the paper. JR contributed to developing the model and edited the paper.

Competing Interests: R. C. O'Reilly is Chief Scientist at eCortex, Inc., which may derive indirect benefit from the work presented here.

Data and Materials Availability: All data and materials will be available at `https://github.com/ccnlab/deep-obj-cat` upon publication.

Supplementary materials

Materials and Methods
Figures S1 - S9
Table S1

Supplementary Materials For: Deep Predictive Learning as a Model of Human Learning

Randall C. O'Reilly, Jacob L. Russin, and John Rohrlich
Correspondence to: oreilly@ucdavis.edu

September 26, 2019

This PDF includes:

Materials and Methods

Figures S1-S?

Table S1

All of the materials described here, including the experimental study, the computational models, and the code to perform the representational similarity analysis, are all available on our github account at: `https://github.com/ccnlab/deep-obj-cat` For the computational models in particular, the most complete understanding can only be had by directly examining the code for the models, as there are a number of details that are not efficiently captured in this suppplementary materials text.

1 Representational Similarity Analysis Methods

The different representations being compared here are:

Leabra: The DeepLeabra (biological model) TE layer representations (specifically TEs = superficial – results are very similar for deep as well).

Bp: The TEs layer representations from the backpropagation version of biological model, including *What, Where* and *What * Where* integration layers, trained with the V1p and V1hp (low and high resolution pulvinar) layers as final output layers, using the time t target pattern from the $t-1$ input (i.e., as a predictive network).

V1: The gabor-filtered representation of the visual input to both of the above models, which was identical across them.

PredNet: Highest layer (4th Layer) of the PredNet architecture.

Expt: Similarity matrix constructed from human pairwise similarity judgments (see *Behavioral Experiment Methods*).

An optimal category cluster can be defined as one that has high within-cluster similarity and low between-cluster similarity. This can be operationalized by the *contrast* distance metric, based on a 1-correlation (*correlation distance*) measure, as the difference between the average within-cluster similarity and the average between-cluster similarity:

$$cd = \langle 1 - r_{in} \rangle - \langle 1 - r_{out} \rangle \tag{1}$$

With distance-like 1-correlation values, this contrast distance should be minimized (it is typically negative), or equivalently the contrast on raw correlation values can be maximized (it is typically a positive number – just the sign flip of distance value). We refer to the positive numbers and maximization here as that is more natural.

Starting with an initial set of clusters, a permutation-based hill-climbing strategy was used to determine a local minimum in this measure: each item was tested in each of the other possible categories, and if that configuration reduced the overall average contrast distance metric across all items, then it was adopted and the process iterated until no such permutation improved the metric. This algorithm can only decrease the number of clusters (by moving all items out of a given cluster), so different numbers of initial clusters can be used to search the overall space.

Figure S1 shows the resulting categories. The Bp model converged on the same cluster state from all starting configurations tested, varying from 5 to 2 initial categories. This is the cluster set

Centroid

1. pyramid
   - banana
   - layercake
   - trafficcone
   - sailboat
   - trex
2. vertical
   - person
   - guitar
   - tablelamp
3. round
   - doorknob
   - donut
3. round cont'd
   - handgun
   - chair
4. box
   - slrcamera
   - elephant
   - piano
   - fish
5. horiz
   - car
   - heavycannon
   - stapler
   - motorcycle

Bp

1. cat1
   - banana
   - layercake
   - trafficcone
   - sailboat
   - trex
   - person
   - guitar
   - tablelamp
   - doorknob
   - donut
1. cat1 cont'd
   - handgun
   - chair
   - slrcamera
   - elephant
   - piano
   - fish
   - car
2. cat2
   - heavycannon
   - stapler
   - motorcycle

V1

1. cat1
   - trafficcone
   - sailboat
   - person
   - guitar
   - tablelamp
   - chair
2. cat2
   - layercake
   - trex
   - doorknob
   - donut
2. cat2 cont'd
   - handgun
   - slrcamera
   - elephant
   - piano
   - fish
   - car
   - heavycannon
   - stapler
   - motorcycle
3. cat3
   - banana

Figure S1: Shape categories used for similarity matrix plots in main paper. *Centroid* shape categories are near-best for both the Leabra model and the Expt results, and fit our visual intuitions about overall shape. *Bp* are reliably optimal for Bp model from all starting points. *V1* are reliably optimal for V1 inputs, and also were best for the PredNet layer 3 representations.

shown in Figure 4a of the main paper, and has an average contrast distance (*acd*) of 0.0838 (this is relatively low because the patterns were overall quite similar). Likewise, the V1 patterns (which were the same across Leabra and Bp models) reliably converged on the same pattern (shown in Figure 4d), with *acd* = 0.2448.

For the PredNet layer3 representations, starting from the V1 categories was already optimal (*acd* = 0.0250 – very low contrast overall), strongly indicating that it did not go beyond the structure present in the input, even though it did not use the V1 gabor filtering used in the Leabra and Bp models (i.e., this V1-level encoding well-captures the structure of the visual inputs in general). The PredNet pixel and layer 0 representations both converged on essentially a single monolitic category with very low acd (0.0160).

For the Leabra TE representations, we found a set of *centroid* shape categories that are near-best when considering both the Leabra model and the results from the human behavioral experiment (Expt). Starting from these categories, the permutation analysis converged on reducing the size of the vertical and round categories to one item each, over a sequence of 5 steps. This is consistent with the observation from Figure 2a that there are three broader categories within which the 5 finer-grained categories are embedded (i.e., vertical and pyramid are overall similar to each other, as are round and box). Nevertheless, our initial visual intuition about the broad shape categories, along with a bias against having single-item categories, reinforced the use of the finer-grained centroid selection. The average contrast difference of our centroid selection is 0.5071, while the maximal result from the permutation was 0.5526, which is a relatively small proportional difference.

Furthermore, once we had collected the human experimental data (*Expt*), it was clear that it strongly coincided with our original shape intuitions, and with the finer-grained 5 category centroid structure. Starting from the centroid categories, the maximal permutation made only 3 changes, moving trex (T-rex) and handgun into the horizontal category, and chair into the pyramid, going from a distance score of 0.3083 to 0.3225, which is a relatively small improvement. However, using the maximal *Expt* clusters directly on the Leabra model gives a lower *acd* measure of 0.3745 (compared to 0.5071 for centroid), so the centroid categories represent a good middle-ground between experiment and the model, and this strong shared similarity structure with near-optimal cluster structures confirms that the model and people are encoding largely the same information.

In contrast, if we organize the experiment similarity matrix using the Bp categories, it produces a very poor average contrast distance measure of 0.0643 (compared to 0.3083 for the centroid categories), strongly suggesting that people's shape representations are not compatible with that simple structure.

Another approach to determining clusters from similarity matrices, *agglomerative clustering*, starts with all items as singletons, and iteratively combines the closest two into a new cluster. The results for the Leabra and Expt similarity matrices are shown in Figure S2, which has also color-coded the items in terms of their category status according to the centroid structure. Due to a strong history dependency in the clustering process, and the indeterminacy of reducing a high-dimensional similarity structure down to two dimensions, structure beyond the leaf level is not very reliable (ties are also broken by a random number generator), but nevertheless you can clearly see that in both cases items from the same cluster are almost always together as leaves in the plots. This then provides additional converging support for the idea that the model is learning the same kind of shape categories as people have.

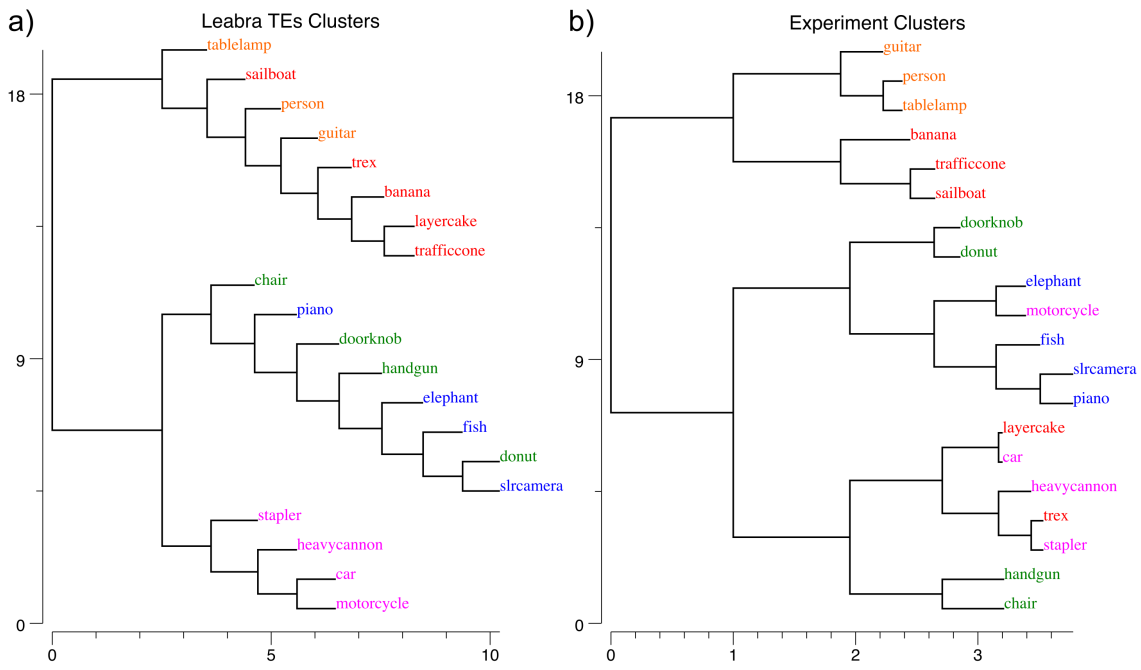

Figure S2: Agglomerative clustering on the Leabra and Expt representations, with the centroid categories color coded. The most reliable information from this is the leaf-level groupings, as the rest of the structure is indeterminante and history dependent in reducing higher-dimensional structure down to a 2D plot. Both cluster plots show a strong tendency to group leaf items together in the same centroid categories, with a few exceptions in each case. Also, the Leabra plot nicely captures the broader 3-category structure evident in the similarity matrix plots, within which the 5 finer-grained centroid categories are organized. Overall, this provides further confirmation that the model and the human subjects are organizing the shapes in largely the same way.

Figure S3: Example stimulus from the behavioral experiment, using the V1 reconstruction of the actual input images presented to the model, to better capture the coarse-grained perception of the model. Subjects were requested to choose which of the two pairs, Left or Right, was most similar in terms of *overall shape*.

2 Behavioral Experiment Methods

The behavioral experiment was conducted on Amazon.com's MTurk web platform under University of Colorado IRB approval (19-0176), using 30 participants each categorizing up to 800 image pairs as shown in Figure S3, using the standard *simple image categorization* framework with a lightly customized script. Objects were drawn from the 156 3D object set, but data was aggregated

in terms of the 20 basic-level categories (car, stapler, etc) because we could not sample all 156 x 156 object pairs. Thus, the resulting data was aggregated for each category pair in terms of the proportion of times when that pair was selected when presented.

The individual images were produced by reconstructing from the V1 transform that the computational model used in its high resolution V1 input layer, to give human participants as similar of an experience as possible to how the model "saw" the objects, and to reduce the influence of existing semantic knowledge which was entirely missing in our model (Figure S3).

3 Biological Model Methods

This section provides more information about the *DeepLeabra What-Where Integration (WWI)* model. The purpose of this information is to give more detailed insight into the model's function beyond the level provided in the main text, but with a model of this complexity, the only way to really understand it is to explore the model itself. It is available for download at: `https://github.com/ccnlab/deep-obj-cat/sims/C++` Furthermore, the best way to understand this model is to understand the framework in which it is implemented, which is explained in great detail, with many running simulations explaining specific elements of functionality, at `http://ccnbook.colorado.edu`

3.1 Layer Sizes and Structure

Figure 1 in the main text shows the general configuration of the model, and Table S1 shows the specific sizes of each of the layers, and where they receive inputs from.

All the activation and general learning parameters in the model are at their standard Leabra defaults.

3.2 Projections

The general principles and patterns of connectivity are shown in Figures S4 and S5.

Detailing each of the specific parameters associated with the different projections shown in Table S1 would take too much space — those interested in this level of detail should download the model from the link shown above. There are topographic projections between many of the lower-level retinotopically-mapped layers, consistent with our earlier vision models *(31)*. For example the 8x8 unit groups in V2 are reduced down to the 4x4 groups in V3 via a 4x4 unit-group topographic projection, where neighboring units have half-overlapping receptive fields (i.e., the field moves over 2 unit groups in V2 for every 1 unit group in V3), and the full space is uniformly tiled by using a wrap-around effect at the edges. Similar patterns of connectivity are used in current deep convolutional neural networks. However, we do not share weights across units as in a true convolutional network.

The projections from ObjVel (object velocity) and SaccadePlan layers to LIPs, LIPd were initialized with a topographic sigmoidal pattern that moved as a function of the position of the unit group, by a factor of .5, while the projections from EyePos were initialized with a gaussian pattern. These patterns multiplied uniformly distributed random weights in the .25 to .75 range, with the

		Units		Pools		
Area	**Name**	**X**	**Y**	**X**	**Y**	**Receiving Projections**
V1	V1s	4	5	8	8	
	V1p	4	5	8	8	V1s V2d V3d V4d TEOd
V1h	V1hs	4	5	16	16	
	V1hp	4	5	16	16	V1s V2d V3d V4d TEOd
Eyes	EyePos	21	21			
	SaccadePlan	11	11			
	Saccade	11	11			
Obj	ObjVel	11	11			
V2	V2s	10	10	8	8	V1s LIPs V3s V4s TEOd V1p V1hp
	V2d	10	10	8	8	**V2s** V1p V1hp LIPd LIPp V3d V4d V3s TEOs
LIP	MtPos	1	1	8	8	V1s
	LIPs	4	4	8	8	MtPos ObjVel SaccadePlan EyePos LIPp
	LIPd	4	4	8	8	**LIPs** LIPp ObjVel Saccade EyePos
	LIPp	1	1	8	8	**MtPos** V1s LIPd
V3	V3s	10	10	4	4	V2s V4s TEOs DPs LIPs V1p V1hp DPp TEOd
	V3d	10	10	4	4	**V3s** V1p V1hp DPp LIPd DPd V4d V4s DPs TEOs
	V3p	10	10	4	4	**V3s** V2d DPd TEOd
DP	DPs	10	10			V2s V3s TEOs V1p V1hp V3p TEOp
	DPd	10	10			**DPs** V1p V1hp DPp TEOd
	DPp	10	10			**DPs** V2d V3d DPd TEOd
V4	V4s	10	10	4	4	V2s TEOs V1p V1hp
	V4d	10	10	4	4	**V4s** V1p V1hp V4p TEOd TEOs
	V4p	10	10	4	4	**V4s** V2d V3d V4d TEOd
TEO	TEOs	10	10	4	4	V4s V1p V1hp TEs
	TEOd	10	10	4	4	**TEOs TEOd** V1p V1hp V4p TEOp TEp TEd
	TEOp	10	10	4	4	**TEOs** V3d V4d TEOd TEd
TE	TEs	10	10	4	4	TEOs V1p V1hp
	TEd	10	10	4	4	**TEs TEd** V1p V1hp V4p TEOp TEp TEOd
	TEp	10	10	4	4	**TEs** V3d V4d TEOd

Table S1: Layer sizes, showing numbers of units in one pool (or entire layer if Pool is missing), and the number of Pools of such units, along X,Y axes. Each area has three associated layers: *s* = superficial layer, *d* = deep layer (context updated by 51B neurons in same area, shown in bold), *p* = pulvinar layer (driven by 5IB neurons from associated area, shown in bold).

lowest values in the topographic pattern having a multiplier of .6, while the highest had a multiplier of 1 (i.e., a fairly subtle effect). This produced faster convergence of the LIP layer when doing *Where* pathway pre-training compared to purely random initial weights. In addition to exploring different patterns of overall connectivity, we also explored differences in the relative strengths of receiving projections, which can be set with a `wt_scale.rel` parameter in the simulator. All feedforward pathways have a default strength of 1. For the feedback projections, which are typically weaker (consistent with the biology), we explored a discrete range of strengths, typically .5, .2, .1, and .05. The strongest top-down projections were into V2s from LIP and V3, while

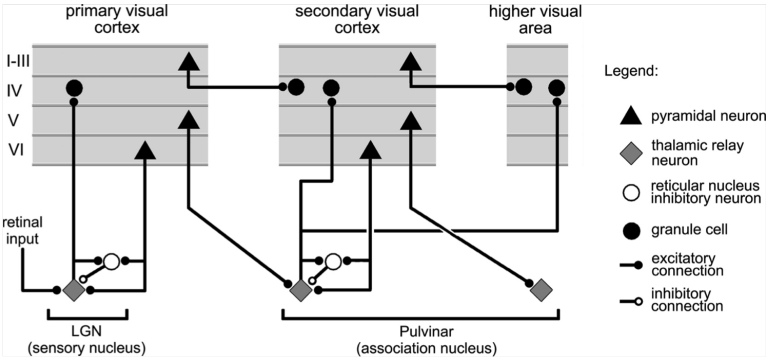

Figure S4: Summary figure from Sherman & Guillery (2006) showing the strong feedforward driver projection eminating from layer 5IB cells in lower layers (e.g., V1), and the much more numerous feedback "modulatory" projection from layer 6CT cells. We interpret these same connections as providing a prediction (6CT) vs. outcome (5IB) activity pattern over the pulvinar. Note that much of the bursting dynamics discussed by these authors is not found in awake behaving animals, where pulvinar neurons exhibit sustained activity over the course of visual inputs similar to what is observed in corresponding visual areas (Robinson, 1993).

most others were .2 or .1. Likewise projections from the pulvinar were weaker, typically .1. These differences in strength sometimes had large effects on performance during the initial bootstrapping of the overall model structure, but in the final model they are typically not very consequential for any individual projection.

3.3 Training Parameters

Training typically consisted of 512 alpha trials per epoch (51.2 seconds of real time equivalent), for 1,000 such epochs. Each trial was generated from a virtual reality environment in the emergent simulator, that rendered first-person views with moving eye position onto the object tumbling through space with fixed motion and rotation parameters over the sequence of 8 frames (see Figure 1c in main text for representative example). Because the start of each sequence of 8 frames is unpredictable, we turned off learning for that trial, which improves learning overall. We have recently developed an automatic such mechanism based on the running-average (and running variance) of the prediction error, where we turn off learning whenever the current prediction error z-normalized by these running average values is below 1.5 standard deviations, which works well, and will be incorporated into future models. Biologically, this could correspond to a connection between pulvinar and neuromodulatory areas that could regulate the effective learning rate in this way.

Figure S6a shows the learning trajectory of the model, indicating that it learns quite rapidly. This rapid initial learning is likely facilitated by the extensive use of shortcut connections convering from all over the simulated visual system onto the V1 pulvinar layers, and direct projections back from these pulvinar layers. Thus, error signals are directly communicated and can drive learning quickly and efficiently. However, there are also extensive indirect, bidirectional connections among the superficial layers, which can drive indirect error backpropagation learning as well.

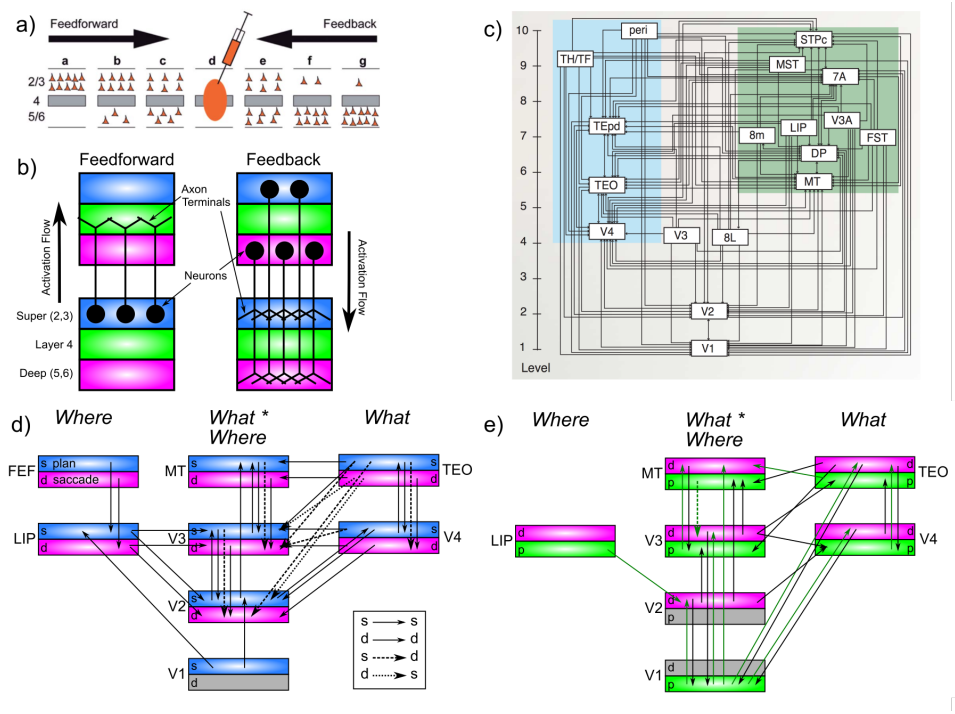

Figure S5: Principles of connectivity in DeepLeabra. **a)** Markov et al (2014) data showing density of *retrograde* labeling from a given injection in a middle-level area (d): most feedforward projections originate from superficial layers of lower areas (a,b,c) and deep layers predominantly contribute to feedback (and more strongly for longer-range feedback). **b)** Summary diagram showing most feedforward connections originating in superficial layers of lower area, and terminating in layer 4 of higher area, while feedback connections can originate in either superficial or deep layers, and in both cases terminate in both superficial and deep layers of the lower area (adapted from Felleman & Van Essen, 1991). **c)** Anatomical hierarchy as determined by percentage of superficial layer source labeling (SLN) by Markov et al (2014) — the hierarchical levels are well matched for our model, but we functionally divide the dorsal pathway (shown in green background) into the two separable components of a *Where* and a *What * Where* integration pathway. **d)** Superficial and deep-layer connectivity in the model. Note the repeating motif between hierarchically-adjacent areas, with bidirectional connectivity between superficial layers, and feedback into deep layers from both higher-level superficial and deep layers, according to canonical pattern shown in panels a and b. Special patterns of connectivity from TEO to V3 and V2, involving crossed super-to-deep and deep-to-super pathways, provide top-down support for predictions based on high-level object representations. **e)** Connectivity for deep layers and pulvinar in the model, which generally mirror the corticocortical pathways (in d). Each pulvinar layer (p) receives 5IB driving inputs from the labeled layer (e.g., V1p receives 5IB drivers from V1). In reality these neurons are more distributed throughout the pulvinar, but it is computationally convenient to organize them together as shown. Deep layers (d) provide predictive input into pulvinar, and pulvinar projections send error signals (via temporal differences between predictions and actual state) to *both* deep and superficial layers of given areas (only d shown). Most areas send deep-layer prediction inputs into the main V1p prediction layer, and receive reciprocal error signals therefrom. The strongest constraint we found was that pulvinar outputs (colored green) must generally project only to higher areas, not to lower areas, with the exceptions of DPp $\rightarrow$ V3 and LIPp $\rightarrow$ V2. V2p was omitted because it is largely redundant with V1p in this simple model.

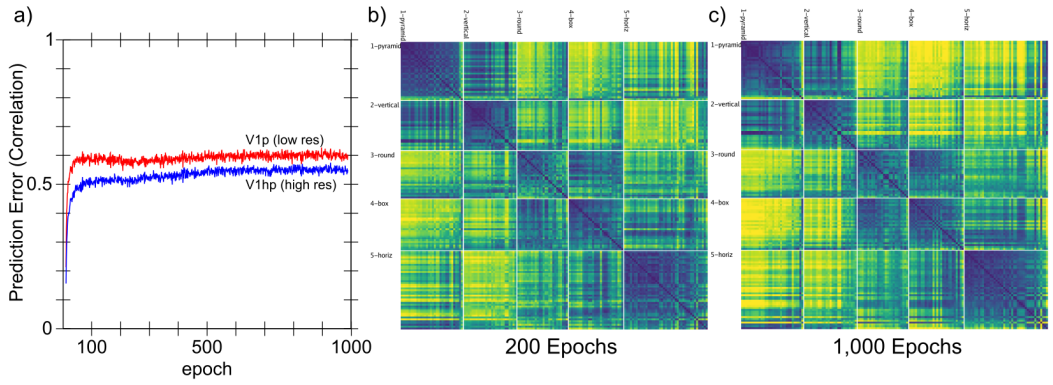

Figure S6: **a)** Predictive learning curve for DeepLeabra, showing the correlation between prediction and actual over the two different V1 layers. Initial learning is quite rapid, followed by a slower but progressive learning process that reflects development of the IT representations (e.g., manipulations that interfere with those areas selectively impair this part of the learning curve). Overall prediction accuracy remains far from perfect, as shown in Figure 1c in main text, and significantly worse than the backpropagtion-based models. This is a typical finding from Leabra models which are significantly more constrained as a result of bidirectional attractor dynamics, Hebbian learning, and inhibitory competition – i.e., the very things that are likely important for forming abstract categorical representations. **b)** Similarity matrix over TEs layer at 200 epochs, which has less contrast and definition compared to the 1,000 epoch result (**c** also shown in Figure 2a in main text).

3.4 Testing Parameters

To be able to monitor similarity metrics as the model trained, we used a running-average integration of neural activity across trials to accumulate the patterns used for the RSA analysis described above. Specifically, for each object, and each frame, the current activation pattern across each layer was recorded and averaged unit-by-unit with a time constant of $\tau = 10$. Critically, by integrating separately for each frame, this running-average computation did not introduce any bias for temporally-adjacent frames to be more similar. Nevertheless, when we computed the frame-to-frame similarities for TE, they were quite high (.901 correlation on average across all objects).

3.5 Model Algorithms

The biologically-based model was implemented using the Leabra framework, which is described in detail in previous publications *(6,12,13,34)*, and summarized here. There are two main implementations of Leabra, one in the *C++ emergent* software, and a new one using Go and Python language at: `https://github.com/emer/leabra`. There are also other simpler implementations in Python and MATLAB, see `https://grey.colorado.edu/emergent/index.php/Leabra`. Both of the preceeding links contain a full detailed description of the algorithm. These same equations and standard parameters have been used to simulate over 40 different models in *(12,13)*, and a number of other research models. Thus, the model can be viewed as an instantiation of a systematic modeling framework using standardized mechanisms, instead of constructing new mechanisms for each model. Here, we only detail properties of the predictive learning algorithm

that go beyond the basic Leabra model.

3.5.1 Deep Context

At the end of every plus phase, a new deep-layer context net input is computed from the dot product of the context weights times the sending activations, just as in the standard net input:

$$\eta = \langle x_i w_{ij} \rangle = \frac{1}{n} \sum_i x_i w_{ij} \tag{2}$$

This net input is then added in with the standard net input at each cycle of processing.

The relative strength of these context layer inputs was set progressively larger for higher layers in the network, with a maximum of 4 in V4, TEO, and TE. In addition, TEO and TE received *self* context projections which provide an extended window of temporal context into the prior 200 msec interval. These self projections were connected only within the narrower Pool level of units, enabling these neurons to develop mutually-excitatory loops to sustain activations over the multiple trials when the same object was present. We hypothesize that these modifications correspond to biological adaptations in IT cortex that likewise support greater sustained activation of object-level representations.

Learning of the context weights occurs as normal, but using the sending activation states from the *prior* time step's activation.

3.5.2 Computational and Biological Details of SRN-like Functionality

Predictive auto-encoder learning has been explored in various frameworks, but the most relevant to our model comes from the application of the SRN to a range of predictive learning domains *(10,11)*. One of the most powerful features of the SRN is that it enables error-driven learning, instead of arbitrary parameter settings, to determine how prior information is integrated with new information. Thus, SRNs can learn to hold onto some important information for a relatively long interval, while rapidly updating other information that is only relevant for a shorter duration. This same flexibility is present in our DeepLeabra model. Furthermore, because this temporal context information is hypothesized to be present in the deep layers throughout the entire neocortex (in every microcolumn of tissue), the DeepLeabra model provides a more pervasive and interconnected form of temporal integration compared to the SRN, which typically just has a single temporal context layer associated with the internal "hidden" layer of processing units.

An extensive computational analysis of what makes the SRN work as well as it does, and explorations of a range of possible alternative frameworks, has led us to an important general principle: *subsequent outcomes determine what is relevant from the past*. At some level, this may seem obvious, but it has significant implications for predictive learning mechanisms based on temporal context. It means that the information encoded in a temporal context representation cannot be learned at the time when that information is presently active. Instead, the relevant contextual information is learned on the basis of what happens next. This explains the peculiar power of the otherwise strange property of the SRN: the temporal context information is preserved as a *direct copy* of the state of the hidden layer units on the previous time step (Figure S7), and then learned synaptic weights integrate that copied context information into the next hidden state (which is then

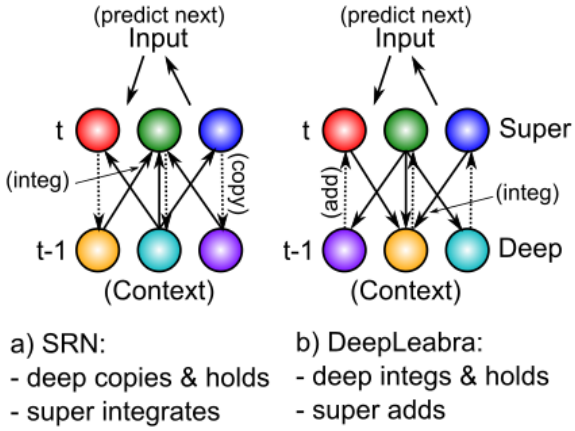

Figure S7: How the DeepLeabra temporal context computation compares to the SRN mathematically. **a)** In a standard SRN, the context (deep layer biologically) is a copy of the hidden activations from the prior time step, and these are held constant while the hidden layer (superficial) units integrate the context through learned synaptic weights. **b)** In DeepLeabra, the deep layer performs the weighted integration of the soon-to-be context information from the superficial layer, and then holds this integrated value, and feeds it back as an additive net-input like signal to the superficial layer. The context net input is pre-computed, instead of having to compute this same value over and over again. This is more efficient, and more compatible with the diffuse interconnections among the deep layer neurons. Layer 6 projections to the thalamus and back recirculate this pre-computed net input value into the superficial layers (via layer 4), and back into itself to support maintenance of the held value.

copied to the context again, and so on). This enables the error-driven learning taking place in the *current* time step to determine how context information from the *previous* time step is integrated. And the simple direct copy operation eschews any attempt to shape this temporal context itself, instead relying on the learning pressure that shapes the hidden layer representations to also shape the context representations. In other words, this copy operation is essential, because there is no other viable source of learning signals to shape the nature of the context representation itself (because these learning signals require future outcomes, which are by definition only available later).

The direct copy operation of the SRN is however seemingly problematic from a biological perspective: how could neurons copy activations from another set of neurons at some discrete point in time, and then hold onto those copied values for a duration of 100 msec, which is a reasonably long period of time in neural terms (e.g., a rapidly firing cortical neuron fires at around 100 Hz, meaning that it will fire 10 times within that context frame). However, there is an important transformation of the SRN context computation, which is more biologically plausible, and compatible with the structure of the deep network (Figure S7). Specifically, instead of copying an entire set of activation states, the context activations (generated by the phasic 5IB burst) are immediately sent through the adaptive synaptic weights that integrate this information, which we think occurs in the 6CC (corticortical) and other lateral integrative connections from 5IB neurons into the rest of the deep network. The result is a *pre-computed net input* from the context onto a given hidden unit (in the original SRN terminology), not the raw context information itself. Computationally, and metabolically, this is a much more efficient mechanism, because the context is, by definition, un-

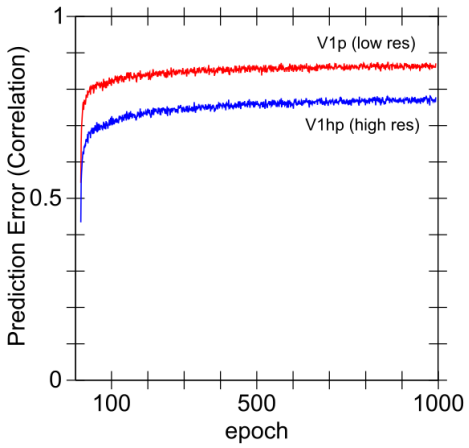

Figure S8: Learning curves for the backpropagation version of the WWI model. Although it achieves better predictive accuracy than the DeepLeabra version, it fails to acquire abstract object category structure, indicating a potential tradeoff between simplifying and categorizing inputs, versus predicting precisely where the low-level visual features will move.

changing over the 100 msec alpha cycle, and thus it makes more sense to pre-compute the synaptic integration, rather than repeatedly re-computing this same synaptic integration over and over again (in the original feedforward backpropagation-based SRN model, this issue did not arise because a single step of activation updating took place for each context update — whereas in our bidirectional model many activation update steps must take place per context update).

There are a couple of remaining challenges for this transformation of the SRN. First, the pre-computed net input from the context must somehow persist over the subsequent 100 msec period of the alpha cycle. We hypothesize that this can occur via NMDA and mGluR channels that can easily produce sustained excitatory currents over this time frame. Furthermore, the reciprocal excitatory connectivity from 6CT to TRC and back to 6CT could help to sustain the initial temporal context signal. Second, these contextual integration synapses require a different form of learning algorithm that uses the sending activation from the prior 100 msec, which is well within the time constants in the relevant calcium and second messenger pathways involved in synaptic plasticity.

4 Backpropagation Model Methods

The backpropagation version of the WWI model has exactly the same layer sizes and *feedforward* patterns of connectivity as the DeepLeabra version. Topographically, the V1p and V1hp pulvinar layers serve as output layers at the highest level of the network, receiving all the various connections from deep layers as shown in Table S1. Likewise, the LIPp served as a target output layer for the Where pathway. To achieve predictive learning, the V1 pulvinar targets were from the scene at time t, while the V1s inputs were from the scene at time $t-1$. We also ran a comparison auto-encoder model that had inputs and target outputs from the same time step, and it showed even less systematic organziation of its higher-level representations, further supporting the notion that predictive learning is important, across all frameworks. The learning curve for the predictive

version is shown in Figure S8, which shows better overall prediction accuracy compared to the DeepLeabra model. However, as the RSA showed, this backpropagation model failed to learn object categories that go beyond the input similarity structure, indicating that perhaps it was paying too much "attention" in learning to this low-level structure, and lacked the necessary mechanisms to enable it to impose a simplifying higher-level structure on top of these inputs.

5 PredNet Model Methods

The PredNet architecture was designed to incorporate principles from predictive coding theory into a neural network model for predicting the next frame in a video sequence. Details of the model can be found in the original paper *(14)*, but here we provide a brief overview of architecture.

5.1 Architecture

PredNet is a deep convolutional neural network that is composed of layers containing discrete modules. The lowest layer generates a prediction of incoming inputs (i.e. the pixels in the next frame), while each of the higher layers attempts to predict the *errors* made by the previous layer. Each layer contains an input convolutional module (A_l), a recurrent representational module (R_l), a prediction module ($\hat{A}_l$), and a representation of its own errors (E_l). The input convolutional module (A_l) transforms its input with a set of standard convolutional filters, a rectified linear activation function, and a max-pooling operation. The recurrent representation module (R_l) is a convolutional LSTM, which is a recurrent convolutional network that replaces the matrix multiplications in the standard LSTM equations with convolutions, allowing it to maintain a spatially organized representation of its inputs over time. The prediction module ($\hat{A}_l$) consists of another standard convolutional layer and rectified linear activation that is used to generate predictions from the output of R_l. These predictions are then compared against the output of the input convolutional module (A_l). The errors generated in this comparison are represented explicitly in E_l, which applies a rectified linear activation to a concatenation of the positive ($A_l - \hat{A}_l$) and negative ($\hat{A}_l - A_l$) prediction errors. These errors then become the inputs to the next layer.

$$A_l^t = \begin{cases} x_t, & \text{if } l = 0 \\ MaxPool(ReLU(Conv(E_{l-1}^t))), & \text{if } l > 0 \end{cases} \tag{3}$$

$$\hat{A}_l^t = ReLU(Conv(R_l^t)) \tag{4}$$

$$E_l^t = [ReLU(A_l^t - \hat{A}_l^t); ReLU(\hat{A}_l^t - A_l^t)] \tag{5}$$

$$R_l^t = ConvLSTM(E_l^{t-1}, R_l^{t-1}, UpSample(R_{l+1}^t)) \tag{6}$$

At each time step in the video sequence, PredNet generates a prediction of the next frame. This is done as follows: first, the R_l is computed for each layer starting from the top of the hierarchy (because each R_l^t depends on input from R_{l+1}^t), and then the A_l^t, $\hat{A}_l^t$ and E_l^t are computed in a feed-forward fashion (becauase each A_l^t depends on input from the layer below, E_{l-1}^t).

All analyses in the RSA were conducted using the representations from the R_l layers.

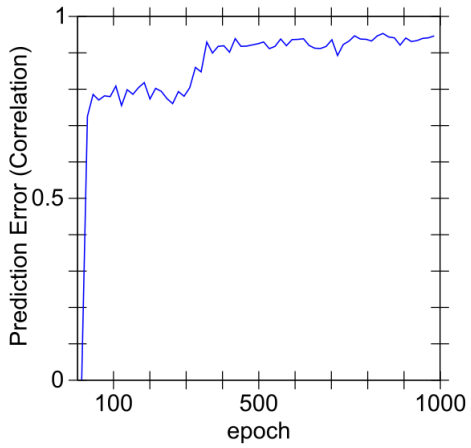

Figure S9: Learning curves for the PredNet model. This model achieves the best overall prediction performance but also has the least well differentiated, categorical representations.

5.2 Implementation details

All experiments with the PredNet architecture were performed using PyTorch. An informal hyperparameter search was conducted, starting from the hyperparameters reported in the original paper. Most hyperparameters reported in the original paper worked best. Our final architecture had 4 layers with 3, 48, 96, and 192 filters in the A_l and R_l modules, and 3x3 kernels throughout the whole network. However, we found that using sigmoid and tanh activation functions in fully-connected convolutional LSTMs slightly improved performance, so these were used for all experiments.

The weights in the PredNet model are trained using error backpropagation. Predictions are generated and errors are computed at all levels of the hierarchy, but the model performs better when only the lowest layer's errors are backpropagated *(14)*. We confirmed these results with experiments that backpropagated the errors in higher layers, in which performance (in terms of mean squared error) was marginally reduced but the RSA results were similar. For this reason, all reported experiments used a PredNet that was trained by only backpropagating the lowest level error.

The model was trained using a batch size of 8 and an Adam optimizer with a learning rate of 0.0001, with no scheduler, for 150,000 iterations. Training curve shown in Figure S9, showing that it achieves the best overall prediction accuracy, and yet has the least well differentiated, categorical representations.