Generalization in neural networks: a broad survey

Chris Rohlfs

Columbia University Department of Electrical Engineering, Mudd 1310, 500 West 120th Street, New York, 10027-6623, NY, USA

ARTICLE INFO

Keywords:
literature review
deep learning
overfitting
causality
domain generalization
transfer learning
foundation models
multimodal
semantic knowledge
abstraction
biologically-inspired

ABSTRACT

This paper reviews concepts, modeling approaches, and recent findings along a spectrum of different levels of abstraction of neural network models including generalization across (1) Samples, (2) Distributions, (3) Domains, (4) Tasks, (5) Modalities, and (6) Scopes. Strategies for (1) sample generalization from training to test data are discussed, with suggestive evidence presented that, at least for the ImageNet dataset, popular classification models show substantial overfitting. An empirical example and perspectives from statistics highlight how models’ (2) distribution generalization can benefit from consideration of causal relationships and counterfactual scenarios. Transfer learning approaches and results for (3) domain generalization are summarized, as is the wealth of domain generalization benchmark datasets available. Recent breakthroughs surveyed in (4) task generalization include few-shot meta-learning approaches and the emergence of transformer-based foundation models such as those used for language processing. Studies performing (5) modality generalization are reviewed, including those that integrate image and text data and that apply a biologically-inspired network across olfactory, visual, and auditory modalities. Higher-level (6) scope generalization results are surveyed, including graph-based approaches to represent symbolic knowledge in networks and attribution strategies for improving networks’ explainability. Additionally, concepts from neuroscience are discussed on the modular architecture of brains and the steps by which dopamine-driven conditioning leads to abstract thinking.

1. Introduction

Generalization is a fundamental objective of deep learning, and recent achievements in the field have expanded the ability of neural network models to consolidate relationships among variables into patterns that apply in other situations. Some such innovations improve the stability and consistency of model performance in a given domain—aims that researchers have emphasized relate directly to a model’s ability to generalize (Bousquet and Elisseeff, 2002)—while other work specifically designs models to adapt to different domains or tasks.

This paper provides a broad overview of different forms of generalization in neural networks. Existing surveys describe theoretical bounds on learning generalizable functions (Mohri et al., 2018), “regularization” techniques for ensuring stability of model forecasts across samples (Bejani and Ghatee, 2021; Kukacka et al., 2018; Shorten and Khosgoftaar, 2019; Qian et al., 2022; Tian and Zhang, 2022) and populations (Liu et al., 2022; Lust and Condurache, 2021; Liu et al., 2023), causal learning to identify counterfactual scenarios (Athey and Imbens, 2019; Guo et al., 2020), methods such as those to adjust when the function of interest is changing (Bayram et al., 2022; Xiang et al., 2023; Yuan et al., 2022), to adapt a model to use an existing skill in a new context (Gulrajani and Lopez-Paz, 2020; Niu et al., 2020; Wang and Deng, 2021; Wang et al., 2023; Zhou et al., 2022; Zhuang et al., 2021), to leverage prior learning to acquire new skills (Li et al., 2021; Vandenhende et al., 2021; Wang et al., 2019c; Zhang and Yang, 2022), the recent emergence of highly versatile transformer-based “foundation” models (Bommasani et al., 2021; Chaudhari et al., 2021), and the representation and extraction of symbolic knowledge in deep learning (Bader and Hitzler, 2005; Besold et al., 2017; Davis, 2017; Ji et al., 2022; Townsend et al., 2020). Studies in the neuroscience literature also discuss methods of abstraction used by biological brains (Devineni and Scaplen, 2022; Mansouri et al., 2020; Meunier et al., 2009). Due to the breadth of topics covered here, this article is in places a review of reviews. It does not provide a comprehensive outline or comparison of state-of-the-art methodologies for any one of these areas but instead offers a high-level view of the diversity of ideas about generalization and approaches to the important question of how and where a neural network can be applied. Its primary aim is to supplement these detailed surveys with an intuitive and concept-driven description of how these types of generalization relate to one another, what type of work is being done in these areas and in some cases, how well the strategies are working, and potential next steps toward achieving in machines the levels of abstraction accomplished by humans and animals.

Figure 1 presents a schematic of different ways in which a model can be generalized. Plotted from left to right is the degree of abstraction involved in the generalization of that model. The left-hand side of the diagram, under the heading “Original,” corresponds to the deployment of a model on its original training data. Moving to the right, the degree of specificity of the application declines, and the level of generality increases. These forms of generalization describe the application of a model across:

1. **Samples**: Out-of-sample test cases drawn from the same population as in training

2. **Distributions**: Out-of-distribution test cases drawn from new populations
3. **Domains**: Out-of-domain contexts where the target function differs from that seen in training

4. **Tasks**: New questions or output classes in the same sphere of knowledge with generally similar input data format

5. **Modalities**: Questions involving fundamentally new input data formats or spheres of knowledge

6. **Scopes**: Complex applications requiring semantic understanding

Authors have used some of these terms in differing and overlapping ways in the literature. Nevertheless, in the absence of a universally accepted lexicon, the seven categories presented in Figure 1 offer a concise taxonomy that captures many of the important ways in which generalization is performed. Discussing them in increasing order of abstraction also helps to highlight ways similarities across these different types of generalization problems.

In the top line of the diagram, just below the two-sided arrow, these six types of generalization are consolidated into three broad categories. The first, stability, includes sample and distribution generalization and describes the ability of models to generalize across minor input perturbations. The ground truth target function that drives the input-output relationships is the same as in training. The only changes are that the specific observations were not seen in training and might differ substantially from those that were. The second, knowledge transfer, encompasses generalization across domain, task, and modality. It describes situations in which the model was trained to learn one target function, and it is reconfigured, retrained, or fine-tuned so that it can be applied to learn a new but related input-output relationship that potentially uses drastically different data. The third broad category includes the furthest-reaching forms of generalization and is placed under the heading of abstraction. In this category, the information gleaned from the model is distilled into concepts with potentially wide applicability.

Each of the four rows of icons in Figure 1 a key element involved in the process of applying a model to a new situation. The icons illustrate the extent to which changes in the level of generality affect the input data, the mathematical tools used for learning and prediction, the outputs that these tools produce, and the degree to which those outputs are accurate.

For the baseline labeled “Original” in the left-hand column, where no generalization is required, the model was designed and calibrated to maximize the exact metric on which it is being evaluated, and we can expect its level of performance to be high.

Moving to the right, the model is deployed in situations that are increasingly different from this baseline. On the left-hand side of the first row, the input data differ only slightly from training. At greater levels of generality, they eventually take on a completely different form. As the second row shows, the model itself is initially unchanged. At moderate levels of generality, the essential components of the model—what is being generalized—are fixed. While some refitting or reconfiguration may occur, it is the model that is intended to remain stable and to perform with some degree of consistency in the face of changing input feeds and output requirements. At the highest level of abstraction depicted in the diagram, however, the precise model structure is discarded in favor of general principles. As the third row illustrates, the output is expected to differ somewhat as the model is applied to new and different types of observations. At small and moderate levels of generality, these exact predictions may differ, but they are drawn from the same set of possible classes. Higher levels of abstraction involve changes in the set of possible output classes and eventually
Generalization in neural networks

in the overall structure of the problem. The icons on the bottom row of the diagram indicate that applying models to situations that are increasingly different from training is expected to degrade performance.

Sample generalization, discussed in Section 2, focuses on the well-known problem of overfitting in deep learning, whereby a model predicts outcomes more accurately in training than on out-of-sample test cases. Neural networks’ strong performance out-of-sample is compared to effectively uninformative theoretical worst-case guarantees, and regularization strategies for maintaining robust performance out-of-sample are discussed. The accuracies of several popular network-based visual object classifiers are compared between ImageNet-V2 and a separately compiled test set. Suggestive evidence indicates that, while training performance has been steadily improving, the amount of overfitting has been large and stable, consistently accounting for misclassification of 20% or more of test cases. Data augmentation strategies are proposed as potential solutions, but this large training-test gap in a well-known classification problem is an area for future research.

Next, Section 3 examines generalization to other distributions. It requires not only effectively forecasting out-of-sample but to cases that may look very different from those in training. It is thus a more difficult problem than sample generalization. Distribution generalization is known to cause difficulties for data-driven models, because the new test cases may have rare combinations of features or sensitivities whose impacts were not learned precisely in training. As with sample generalization, theoretical worst-case guarantees on out-of-distribution learnability are mostly uninformative. Techniques from machine learning and from classical statistics and econometrics are discussed for one important form of out-of-distribution generalization—to counterfactual cases for causal inference. As a motivating example, a counterfactual image is presented of a lion on a city block. While the true class of the image is readily apparent to humans, some popular models are found to misclassify it. Effective generalization to counterfactual cases is highlighted as a promising area for future deep learning research that draws considerable interest from applied practitioners.

Out-of-domain generalization, covered in Section 4, is in a sense an extreme version of out-of-distribution generalization. The model is applied to a new situation with modified rules for mapping input features to output classes—for instance, identifying objects from cartoons rather than photos or classifying documents written in Icelandic rather than English. A cartoon or an Icelandic text would be a rare or impossible case in the training data of photographs or English texts. The relationship between inputs and output classes is distorted so much from that seen in training that it is most practical to model it as its own distinct target function. The section covers two forms of generalization across domains. First, the problem of “concept drift,” or a gradual change in domain, is reviewed, along with established strategies for addressing it. Case studies are discussed, including the failure of Google Flu Trends and Completely Automated Public Turing tests to tell Computers and Humans Apart (CAPTCHA). Second, datasets, approaches, and results are discussed for sharp and observable shifts in domain. The photo-Art-Cartoon-Sketch (PACS) visual classification exercise—in which the network is trained to recognize the same objects in different visual vernaculars (four different domains)—is considered as a motivating example. Domain generalization begins with an established “foundation model” that is pretrained on a well-known dataset of images and used as a “backbone” and whose parameters are held fixed. New network layers are added to perform the classification task at hand, some of which are domain-specific, and strategies known as transfer learning are used to enable the system to classify test cases from a domain that was not seen during training.

Section 5 considers recent developments in the ability to generalize deep learning models to new tasks with different outcome classes than in the original problem. The distribution of cases and the domain (for instance photographs or text) may be unchanged from training. Thus, unlike in distribution or domain generalization, the processing of data features in task generalization is potentially unchanged. Unlike with distribution or domain generalization, however, no prediction is possible without reconfiguring the model, because the structure of the outputs has changed. Early efforts built task flexibility into the network training process and worked to counteract the phenomenon of catastrophic forgetting through which learning a new task would degrade the model’s performance on previously tasks. Recent progress in computer vision is discussed that builds upon the transfer learning approach. A pre-trained image classification backbone is extended with additional layers or steps to perform few-shot learning. These additional elements enable the model to recognize new classes of images after seeing just one or five example cases. Next, transformers are explored in the context of natural language. Networks with this innovative architecture are pre-trained on extensive corpora of text-based data. Like the backbones of image classifiers, these pre-trained transformers serve as foundation models for a variety of tasks. Models built on this common structure are found to perform at or above human level in batteries of language tests. The recent rise of highly versatile transformer-based foundation models is briefly discussed.

In cross-modality generalization, patterns learned from one form of data—for instance, images, text, or audio—are applied to make inferences relevant to another of those areas. Section 6 reviews some recent innovative studies in the literature that generalize models across widely different problems and data types. As with task generalization, it is necessary in modality generalization to reconfigure the model in order for it to produce forecasts. Generalization across modalities can be thought of as an extreme change in domain, so that not only has the target function changed, but there has been a complete change in the structure of the input
2. Sample Generalization

2.1. Bias and Variance

One key variant of generalization that is used with nearly all deep learning models is the application of a trained model to new test cases. Across many areas of empirical research, models are known to forecast more accurately among in-sample observations used in the fitting process than on previously unseen out-of-sample observations, and the issue is particularly severe among complex models with large numbers of estimated parameters. This phenomenon is illustrated in Figure 2, which is modified slightly from Hastie et al. (2009) and shows training error in blue and test error in red, both as functions of model complexity. The image, constructed for illustrative purposes, was constructed by applying LASSO regression to simulated data.

On the left-hand side, the algorithms exhibit a bias toward overly smooth and simple functions that are “underfitted” and omit key predictors and nonlinearities, resulting in high errors in both the training and test samples. This effect is sometimes called “inductive bias,” which refers to the modeling assumptions necessary to make out-of-sample inferences (Mitchell, 1997).

The problem of underfitting, through which an overly simple model does not effectively respond to data, is analogous to the biological phenomenon of fixed action patterns observed in insects and birds. Davies et al. (1989) note that the parenting behaviors of some ants and birds are evolutionarily preprogrammed and thus fixed over the animals’ lifetimes. Their inability to adjust in response to outside stimuli like the characteristics of their offspring leaves them vulnerable to brood parasites. They are consequently tricked into expending resources raising the young of other species.

Mammals, with their greater cognitive resources, do not generally rely upon elaborate instinctive programs of behaviors that are locked down at birth. Instead, each generation must acquire skills like nesting through dopamine-driven learning (Dewsbury, 1978). While costly, this ability to learn and to respond to external events helps them to avoid repeat mistakes.

It is this form of learning that artificial neural networks seek to imitate. By incorporating parameters into the model that are sensitive to situation-specific factors—the input data—the model is trained to behave differently depending upon what features a given out-of-sample test case exhibits.

Moving to the right on the graph, the number of model parameters rises, and this source of error declines. While training error declines consistently with model complexity, test error falls initially and then rises due to variance caused by overfitting—that is, some spurious predictors reduce training error, but their effects do not generalize out-of-sample. Due to this second source of error, which increases with model complexity, a model that is selected to minimize in-sample error is likely to be overly complex relative to a model that minimizes the error from out-of-sample forecasting.

2.2. Conceptual Framework

Despite these risks of under- and overfitting, theoretical work in machine learning provides conditions for achieving a certain type of generalization across samples known as probably approximately correct (PAC) learning. A learning algorithm is PAC for some thresholds δ and ε if fewer data, the output format, or both. One area in which cross-modality generalization has been particularly influential is the intersection of vision and language. Image classifiers trained on caption-enhanced images are found to identify objects more effectively than those trained on images alone.

The most abstract form of generalization, covered in Section 7, is across scopes. It is less about running a new model and more about the interaction between machine and human logic. Through it, models learn and communicate semantic knowledge and general principles, including those that might be represented through decision trees or symbolic logic. This section examines how human knowledge is codified into graphs and incorporated into network-based architectures. An overview is presented of competitions that evaluate the degree to which the resulting graph-enhanced neural networks can reason through open-ended questions.

The topic of knowledge extraction is also covered, with example cases illustrating ways of analyzing the reasoning behind deep learning systems predictions. Both forms of generalization—the representation of semantic knowledge in neural networks and the distillation of learned knowledge into understandable principles—are promising areas for further exploration.

Next, Section 8 reviews neuroscience research on the modular and dopamine-driven means by which abstraction occurs in animal brains. This biological perspective provides insights into modular elements of existing artificial neural networks as well as potential directions in which artificial systems might generalize more effectively.

Section 9 summarizes, discusses potential future directions for network-based abstraction, and concludes.
than $100 \times \delta\%$ of unseen cases have a probability greater than $\epsilon$ of being misclassified. In this framework, $1 - \delta$ is called the confidence level and $\epsilon$ is the error rate. For the examples presented in this section, $\delta = 0.01$ and $\epsilon = 0.05$. This definition allows for some cases with extremely high probability of misclassification, provided that they make up less than 1% of the population. For the remaining 99% of cases, the likelihood of misclassification is contained at 5%.

The number of training cases required to achieve this PAC standard of generalizability depends upon the function being learned, the distribution of cases in the population, and the algorithm. Vapnik (1995) and Mohri et al. (2018) review bounds that have been derived for the minimum sample sizes required for PAC learning, and Table 1 presents some example values. The “Lower Bound” column shows the lowest sample size at which PAC learning is possible for some target function, data distribution, and algorithms. It is calculated as the algorithm’s Vapnik-Chervonikis (VC) dimension divided by $320e$. The “Upper Bound” column shows the sample size at which PAC learning is guaranteed for a broad set of possible target functions, data distributions, and algorithms. It is calculated as the sample size $m$ that numerically solves the nonlinear equation $me^{2}/8 = d \log(2em/d) + \log(4/\delta)$, where $d < m$ is the VC dimension and the mathematical constant $e$ appears in the equation and as the base for the log. While Kawaguchi et al. (2022) derive tighter bounds for specific types of neural networks, the more general and well-known bounds are considered here.

The VC dimension is a measure of model complexity that is the machine learning analogue to degrees of freedom in classical statistics. It is the maximum number of training cases that an algorithm can “shatter,” that is, perfectly fit any set of labels applied to those cases. Imagine, for instance, a spam filter with possible labels of “spam” and “not spam,” with a training sample of 50 emails. There are $2^{50}$ possible ways that these emails could be sorted into the two bins, each reflecting some hypothetical user’s view of which emails should be kept and which should be filtered out. A learning algorithm with a VC dimension of 50 could perfectly fit this 50-observation training set, regardless of which of these quadrillion users labeled the emails. Given 51 emails, however, the algorithm lacks the expressive power to exactly match at least one of the $2^{51}$ quadrillion hypothetical users’ labeling schemes.

While neural networks’ VC dimensions are typically not known, the popular models all have some training error, which demonstrates that they cannot shatter sample sizes as large as their own training sets. The influential LeNet-5 model for classifying handwritten digits (LeCun et al., 1998) uses 60,000 training cases and has some training error when only 15,000 training cases are used. Thus, its VC dimension must fall below 15,000. Additionally, all known classifiers of the ImageNet-V2 (Russakovsky et al., 2015) fail to shatter its 1.28 million-observation training sample and must have

| VC Dimension | Lower Bound | Upper Bound |
|--------------|-------------|-------------|
| 10           | 13          | 13.1 million |
| 100          | 125         | 126 million  |
| 1,000        | 1,250       | 1.26 billion |
| 10,000       | 12,500      | 12.6 billion |
| 100,000      | 125,000     | 126 billion  |
| one million  | 1.25 million| 1.26 trillion|

Table 1
Upper and Lower Bound Sample Sizes for PAC Learning when $\delta = 0.01$ and $\epsilon = 0.05$

VC dimensions lower than that amount. Zhang et al. (2017, 2021) also show that some well-known neural networks have high but not perfect training accuracy on ImageNet-V2, even when the labels are randomly assigned. On the lower end, classifiers are unlikely to have VC dimensions lower than the number of classes (10 for LeNet-5 and 1,000 for ImageNet-V2). Additionally, while they do not establish shattering in the sense of capturing every possible labeling, Zhang et al. (2017, 2021) find that some sophisticated networks can perfectly predict random labels applied to the 50,000-observation CIFAR-10 training set (Krizhevsky, 2014).

As Table 1 shows, the upper bound sample sizes at which PAC learnability is guaranteed are roughly a million times as large as the lower bound sample sizes at which some functions are PAC learnable. Thus, even if a network’s VC dimension were known exactly, there is still a wide range of possibilities for the number of training observations required to learn a given empirical problem. Target functions that are easier to learn, given the algorithm and data distribution, require sample sizes near the lower end of the range, and functions that are harder to learn require sample sizes near the higher end.

One might reasonably expect common machine learning problems, which many humans can solve with little effort, to be easier to learn than random or worst-case groupings of elements, as noted and observed empirically by Kawaguchi et al. (2022). Hence, we might expect them to require sample sizes in the lower part of the range. Indeed, LeNet-5’s error rate on unseen test cases is below 1%, even though its 60,000-observation training set is considerably lower than the millions or billions that the upper bound would require.

2.3. Empirical Strategies

Because highly complex neural networks generalize so well relative to the upper bound for PAC learnability, deep learning research makes extensive use of models that are “overparameterized”—that is, those whose model complexity far exceeds the number of training observations. As in the the right hand side of Figure 2, the training error is particularly low, but the model may suffer from

---

1This is the “unrealizable” case in which the hypothesis set does not necessarily include the true target function but may include a sufficiently close approximation.

2For certain types of neural networks, Bartlett et al. (2019) provide tighter bounds on the VC dimension based upon models’ numbers of parameters.
substantial degradation in performance when generalizing out-of-sample due to the variance that arises from overfitting.

The strategies that neural network researchers use to mitigate this risk are described as “regularization” and are described in detail in surveys by Bejani and Ghatee (2021); Bishop (2006), Goodfellow et al. (2017), Dos Santos and Papa (2023), Kukačka et al. (2018), and Moradi et al. (2020). The approaches can be grouped into three categories: simplifying the model, coarsening the fit, and augmenting the data. The list below describes popular techniques within each of these categories:

(1) **Simplify Model.** Useful for improving generalizability while also reducing computational cost and increasing explainability.

- **Multi-objective Optimization.** Algorithm modified to balance training error and model complexity, often measured as magnitude of parameter weights. Frequently implemented through constrained optimization. Examples in de Alburquerque Teixeira et al. (2000); Rocha et al. (2020) and Torres et al. (2022) with a helpful survey by Tian and Zhang (2022).

- **Weight Sharing.** Reuse weight values across the network to reduce parameters and enhance generalization. Common in convolutional (LeCun et al., 1998) and recurrent networks (Bengio et al., 1994), language translation (Press and Wolf, 2017; Iinan et al., 2017) and autoencoders (Vincent et al., 2010).

- **Pruning.** Eliminate neurons or connections with little influence post-fit (Cheng et al., 2023; Li et al., 2023; Marinó et al., 2023).

- **Knowledge Distillation.** Train a smaller model to mimic a larger one (Hinton et al., 2015; Gou et al., 2021).

(2) **Coarsen Fit.** Helps to concentrate training on broad patterns, which can be most reliably and robustly learned, rather than on small fluctuations.

- **Noise.** Add random values to gradients or parameters to reduce relative importance of minor perturbations (Neelakantan et al., 2015; Blundell et al., 2015).

- **Dropout.** Temporarily drop some neurons during each training iteration to reduce overreliance on individual neurons (Hinton et al., 2012; Srivastava et al., 2014).

- **Batch and Layer Normalization.** Demean and scale inputs within minibatches and across layers to limit influence of artificial sampling variation on parameter updates (Ioffe and Szegedy, 2015).

- **Early Stopping.** Stop training once validation error increases, even if training error continues to decline (Prechelt, 2012).

- **Double Descent.** Overparameterizes models to achieve better generalization, with empirical evidence supporting improved out-of-sample performance at very high levels of model complexity (Belkin et al., 2019; Kawaguchi et al., 2022; Nakkiran et al., 2021; Neyshabur et al., 2017).

(3) **Augment Data.** Useful for addressing specific known forms of overfitting but increases training cost.

- **Input Adjustment.** Add noisy, shifted, or transformed versions of training data to enhance robustness against common classification errors (Bishop, 2006; Shorten and Khoshgoftaar, 2019).

- **Adversarial Learning.** Perturb inputs in worst-case directions to create challenging training cases (Akhtar and Dasgupta, 2019; Chakraborty et al., 2018; Goyal et al., 2023; Qian et al., 2022).

- **Multi-task Learning.** Train the model on more than one related task. For instance, one task might involve identifying when an object or recognizable sound is in frame, while another would determine which object or sound it is, with network elements shared for steps like edge detection that both tasks require (Vandenhende et al., 2021; Zhang and Yang, 2022).

These three categories of regularization approaches tackle the problem of overfitting in very different ways. The first, model simplification, has a straightforward analogue to classical statistics, in which overfitting is directly related to the number of free parameters. It applies the intuition from Figure 2, reducing the number of parameters—or the magnitudes of those parameters—as a means of improving generalizability.

The second category, by contrast, is specific to the mechanics of complex machine learning systems without unique or exact solutions. Approaches to coarsen the fit are born out of experimentation with those systems.

The final category, data augmentation, most closely resembles strategies used in biological learning. As Sinz et al. (2019) notes, data collection in biological systems is curiosity-driven, focusing on training cases that deviate from expectation or fill specific gaps in understanding. By nudging inputs in worst-case directions, adversarial learning employs a similar strategy in the artificial network setting. Sinz et al. (2019) also observes that biological brains divide into modules, and each component of the system is informed by observations from a variety of applications in which it is used. Because of this design feature, biological learning is relatively insensitive to aberrations in any one domain of knowledge. The regularization strategy of multi-task learning applies a similar insight to artificial networks.

### 2.4. Application to ImageNet

Next, to better understand the empirical relevance of overfitting, we turn to an empirical application from the field of computer vision. Specifically, we compare the training, validation, and test performance of twelve established neural networks on the ImageNet-V2 task of identifying objects from a set of 1,000 possible categories. Many of the original studies do not report training accuracy. For the purposes of this discussion and following convention in the literature, overfitting is measured here as the difference between training accuracy and test accuracy (Goodfellow et al.,
The 12 neural networks considered all use many of the regularization techniques described in the previous subsection. All use multi-objective optimization in the form of norm-based penalties for complexity, and all manage model complexity further by employing convolutions and other strategies for weight-sharing. Additionally, all 12 perform dropout and batch normalization to coarsen the fit. Some also adjust the learning rate across training iterations in a way that resembles early stopping, and some achieve the high levels of complexity that has been found to produce the “double descent” phenomenon. Additionally, all augment the training data with transformed versions of the images, including flipping, shifting, cropping, and modifying the colors.

To match the layout of Figure 2, Figures 3a, 3b, and 3c present these performance statistics as the fraction that are incorrectly classified, labeled as “Prediction Error.” If \( A \) is the number of correct predictions divided by the total, then \( 1 - A \) is the value plotted along the vertical axes. As with Figure 2, this error is plotted against model complexity on the horizontal axis. In Figure 3a, the model complexity is measured as the number of parameters. Following Bartlett (1997), Figure 3b measures complexity as the magnitude of the estimated weights, computed here as the Euclidean norm. In Figure 3c, required computations, measured as Giga-Multiply-Accumulate (GMAC) is used as the measure, which is a variation on Floating Point Operations (FLOPs) that is calculated using the fvcore profiler in Python (Yang et al., 2020).

The model names appear next to their levels of performance in the training data.

The same findings from Figure 3 also appear in Table 2. The models are listed in table in decreasing order of publication date. For each of these models, the table presents the source, the values for both measures of complexity, a brief description of the novel elements introduced by that network, and the performance data expressed in terms of percent accurate, or 100 ∗ \( A \). The key measure of overfitting, the difference between training and test accuracy, is shown in the right-hand column. The key finding from Figures 3a and 3b and Table 2 are that, despite the many regularization approaches used, the amount of overfitting, is substantial. The magnitude of this gap, shown in the right-hand column of the table and illustrated graphically as the difference between the red and blue lines, ranges from 19.6 to 32.6 percentage points. In 11 of the 12 models considered here, this gap from overfitting is larger than the training error, GoogLeNet being the one exception.

Across these 12 neural networks, there are suggestive patterns of increased training accuracy and increased norm-based complexity over time, with values of both variables...
generally higher for entries at the top of the table than at the bottom. No such pattern can be seen for overfitting, however. Thus, the evidence from these 12 models suggests that progress in ImageNet accuracy over this period has been effectively independent of overfitting. While training error has declined considerably, the preliminary findings from this comparison indicate that overfitting has been stubbornly persistent as a sizeable problem, and there is not a clearly discernible relationship between the amount of overfitting and model complexity.

These models already employ many of the regularization strategies typically applied to avoid overfitting. While it is unclear exactly why the amount of test error is so consistently large, one possible explanation is that the training data are insufficiently diverse. Augmenting the data with a wider variety of cases—particularly unusual-looking and counterfactual examples that would help with out-of-distribution generalization—is one possible avenue that is worth considering for addressing this issue.

Another pattern visible from Figures 3a and 3b and Table 2 is the large and stable difference between validation and test error. This difference, seen in the graphs as the gap between the green and red lines, ranges from 11.8 to 13.6 percentage points. This evidence indicates that the overfitting seen in popular ImageNet classifiers is only partially attributable to overfitting of the model weights in the training data. This preliminary evidence is consistent with a roughly equal amount of variance coming from network architecture, hyperparameter selection, and other broad judgments made during the design and validation of the model.

3. Distribution Generalization

3.1. Broadening Out-of-Sample Performance

In many situations in practice, researchers wish to generalize models more extensively than simply to randomly selected test samples. In medical settings, it is often desirable to use a single model to diagnose multiple types of patients and to maintain a high standard of accuracy not just for patients on average but separately for each type. Market researchers may want to forecast consumers’ responses to a previously unseen product or advertisement. Educators might try to predict how expanding a pilot program to additional classrooms or schools would affect student outcomes. Applications of the model to subpopulations or to rare or hypothetical cases in this way are forms of out-of-distribution generalization.

In all of these contexts, the target function $f$ that maps features to labels remains unchanged. One of those specific or hypothetical test cases could in principle appear in the training data and would have the correct label applied to it. A random test sample could possibly include it as well, but that case would be one of many and would appear with low probability. What differs from the problem of sample generalization is that there is a particular set of test cases that are of interest, and they are not randomly drawn from the same population as the training sample is.

If the target function $f$ and all the relevant predictors were known with absolute certainty, the researcher could accurately classify any test cases, regardless of the distribution from which they are drawn. A key difficulty arises in learning because the function itself is an approximation that is sensitive to the proportions in which different types of cases appear. The definition of PAC learning allows for hypotheses that have blind spots in the sense that a PAC-learned classifier might perform terribly on some fraction $\delta$ of test cases. No performance guarantee would apply to a test sample that consists solely of this subset of hard-to-classify cases.
To illustrate a key problem that this form of generalization poses, Figure 4 presents example in- and out-of-distribution cases that could be presented to a object recognition classifier. Both images were artificially generated using DALL-E from OpenAI (OpenAI, 2024a,b). Both clearly depict lions, but the first shows one in its natural habitat, while the other presents one in an urban setting. Moving the animal from the savannah does not change the ground truth class to which it belongs, and any human looking at the images would recognize both as lions. In practice, however, the ImageNet training images of lions come almost exclusively from natural and semi-natural settings like wildlife reserves and zoos. Many artificial network-based classifiers rely upon environmental cues like grass and dirt in the surroundings to inform their predictions, and the change in background can throw them off.

Table 3 presents the labels that each of the 12 artificial neural network models from Table 2 apply to the two images in Figure 4, along with the level of confidence to which they assign those labels. As the table shows, all 12 of the artificial neural networks listed in Table 2 correctly identify Figure 4a as a lion, and they generally do so with high confidence. Only half, however, correctly label the image in Figure 4b as a lion. The other six models are evenly split between labeling the animal as a chow chow or a Tibetan mastiff, two dog breeds that are much more likely than a lion to be seen on a city street. While not shown in the table, nearly all of the models that misclassify Figure 4b do include lion in their top five guesses. The sole exception is MobileNet v3 small, whose top five are all dog breeds.

In related work using a Recurrent Neural Network (RNN) to classify handwritten digits, Horoi et al. (2020) find a high degree of sensitivity to slight modifications in test cases. The authors find that, when rows are deleted from a test image, when the last rows are blacked out, or when extra blank rows are added to the sequence, the model projections are inaccurate and unstable. The authors additionally find that the intermediate outputs generated from successive iterations of the recurrence exhibit volatile behavior and do not follow intuitive patterns from one iteration to the next.

In addition to examining specific ways of manipulating test cases, as in Table 3 and the digit-classification exercise, some work takes a systematic approach to the problem of generalization across distributions. A survey by Lust and Condurache (2021) also highlights the use of adversarial examples as a promising means of identifying the limits of a model to generalize out-of-distribution. Such studies generate different types of adversarial cases to trace out the “generalization envelope” describing the scope of distributions over which the model can be expected to perform consistently.
Generalization in neural networks

To assess the ability of learned systems to generalize across distributions, another area of research applies reweighting and subsampling to the training data to simulate potential distributions of test cases. This approach might not help in the classification of hypothetical test cases like Figure 4 or a never-before-seen ad or product, which have negligible likelihood of appearing in training data. They may, however, help to understand performance across subpopulations, as in the medical diagnosis example described above. In one approach known as Distributionally Robust Optimization (DRO), researchers train model parameters to ensure a minimum level of performance in worst-case scenarios. Liu et al. (2023) note in a useful summary that explicit distribution-aware optimization strategies of this form have been found to be impractical due to their high computational demands and sensitivity to underlying assumptions about the potential test sample distributions.

Beyond these strategies for assessing a model’s applicability across potential out-of-distribution test cases, researchers in some cases have explored the nature of the errors that arise in such contexts. Next, we will consider ways in which the data and model structure can be built out to improve deep learning models’ abilities to produce accurate projections across different distributions of test cases.

### 3.2. Counterfactuals and Causality

The generalizability of neural network models to out-of-distribution cases is not unlike the identification of counterfactuals in classical statistics. Consider, for instance, a deep learning model that uses data on patients’ medical histories to project their likelihoods of heart disease. If sufficiently many training cases are used, learning theory guarantees that the predictions of heart disease will be PAC for test cases drawn at random from that same population. What if, however, a doctor is considering advising patients to stop smoking? The doctor’s advice effectively envisions counterfactual out-of-distribution test cases. The hypothetical patients that the doctor imagines have histories of smoking, and they exhibit whatever behaviors and genetic, socioeconomic, and psychological factors are correlated with smoking in the data. Due to the doctor’s intervention, however, they do not smoke.

A model that captures general patterns in the data might overstate or understate the importance of smoking relative to other risk factors like cholesterol intake and sedentary lifestyle—and the direction and magnitude of that error is likely to vary across patients in ways that are hard to predict. Such imprecision will not cause major problems in conventional out-of-sample forecasting, because smoking tends to appear along with those other risk factors in the general population. For in-distribution purposes, smoking, cholesterol, and lifestyle are good enough proxies for one another. As noted in Section 2, coarse fitting procedures that focus on general patterns tend to generalize well in-distribution. When the researcher wishes to apply the model to out-of-distribution counterfactual cases, however, this failure to correctly isolate each predictor’s contribution becomes important.

At the heart of this issue are confounding variables, features that co-move with the predictor of interest and also have their own impacts on the model outcome. The difficulty in inference that these variables cause is known in statistics and econometrics as omitted variables bias. It is this same problem that prevents the 12 popular neural networks from recognizing a lion in the city. If a photo of a lion is drawn from the same distribution as was used for training, it will include the natural-looking surrounding environment that one would expect from the training data. The pattern in which lions always appear together with grass and dirt is only broken by the appearance of a counterfactual, out-of-distribution test case.

As the lion in the city example shows, being able to observe all the relevant data does not necessarily solve the problem of identifying counterfactuals. In that situation, the direct inputs to the model are pixel values, and the confounding variables—descriptors of the background environment—are complex combinations of those pixel values. Those confounding features could potentially be learned in the deep layers of the model. Training data do not, however, typically include cases that shift those features independently of the objects being recognized. For this reason, we should not expect models to learn how to disentangle deep confounders from the ground truth objects they seek to identify.

Even if it is observed directly, as might be the case with a predictor like cholesterol intake in the smoking example, a confounding variable could still get in the way of forecasting counterfactual outcomes. Due to the coarseness of many neural networks’ fits and the lack of independent variation in the predictor of interest—in this case smoking cessation—a deep learning model may fail to accurately separate the effect of that variable from that of the confounder. Additionally, many of the confounding variables, like the many genetic, socioeconomic, and psychological determinants of smoking, can only be measured with rough proxies. For cases like health outcomes in which key drivers are unobservable, this problem of confounding variables and a lack of out-of-distribution generalizability is inevitable.

Given the practical value of counterfactuals for guiding policy decisions, business strategies, and other real-world interventions, some researchers have begun to explore the identification of causal relationships learning-based approaches. In addition to having these direct applications for evaluating potential courses of action, causal links are often tied to physical, biological, or economic processes. Many argue that such fundamental relationships are more stable and robust than statistical associations, which can be especially sensitive to differences across populations and over time—and thus more reliable for generalization across populations and distributions (Pearl, 2009; Rubin, 1974; Schölkopf et al., 2021).

Outside of machine learning, statisticians and econometricians have developed a variety of approaches for identifying counterfactual scenarios and causal relationships. Randomized Controlled Trials (RCTs) are widely agreed to be the most reliable. By directly choosing who receives a
treatment, the researchers can ensure that it is independent of any potential confounding factors. RCTs are often costly and raise ethical issues, and so another popular strategy involves “natural experiments” or “quasi-experiments”: events that caused some essentially random set of subjects to receive a treatment, so that the experimental environment can be simulated using observational data. This essentially random driver of treatment is called an instrumental variable. In the example of quitting smoking, regulations such as cigarette taxes are sometimes used as instrumental variables. DeCicca et al. (2022) provides a review. For situations in which RCTs and quasi-experimental strategies are not viable but key confounding variables are observed directly, researchers sometimes use regressions or matched-pair comparisons to separate their effects from those of the predictor of interest. The use of “control variables” in this way is referred to by Pearl (2009) as a “back-door” adjustment. Popular early references for RCTs, quasi-experiments, matching, and other methods for causal inference include Campbell and Stanley (1963); Cook and Campbell (1979), and Rubin (1974). Angrist and Fischke (2009) and Pearl (2009) provide more recent overviews.

Research on causality and counterfactuals is not as well developed for learning-based settings as it is in classical statistics. One area that has been explored in the learning-based estimation of causal relationships is “structural” graph-based models. Popular strategies of this form are built around Directed Acyclic Graphs (DAGs). Predictors are ordered into tiers based upon their relative positions in the causal chain, which is then used to inform the network structure. Orderings are obtained from domain knowledge or through adversarial learning, based on the premise that relationships that are robust to adversarial attacks are likely to be causal. Kaddour et al. (2022); Peters et al. (2017); Schölkopf et al. (2021); Vowels et al. (2022), and Zanga et al. (2022) provide useful surveys. Such structures may help to rule out some highly implausible effects, such as tomorrow’s outcomes impacting events today. They do not, however, address the far more serious challenge of unobserved confounding variables, and they are generally not tested for out-of-distribution generalization.

Counterfactuals and deep learning both spark intense interest among practitioners, and the intersection of the two is a ripe area for future research. In addition to graph-based strategies, some promising recent studies exist combine established methods for identifying counterfactuals with learning-based approaches for capturing complex, nonlinear relationships with rich sets of interaction effects. Summaries can be found in Guo et al. (2020); Athey and Imbens (2019) and Chernozhukov et al. (2024). While few combine neural networks with experimental or quasi-experimental designs, one notable exception is Hartford et al. (2017). That paper applies an deep learning instrumental variables approach to simulated data and to a real world application of online ads. In the application, the researchers find that the network-based approach produces similar findings as an earlier parametric study but with less manual effort due to automated feature selection. In another interesting study, van der Laan and Rose (2011) describe an ensemble-based “targeted learning” approach through which shallow or deep learning models can help to refine the estimated treatment effects obtained from RCTs and increase the level of precision with which impacts can be estimated. Pirracchio et al. (2020) presents an application that combines shallow learning models with RCTs to assess a corticosteroid treatment.

Another area of interest in causal machine learning is matching. Chernozhukov et al. (2018) and Wager and Athey (2018), for instance, combine shallow learning with propensity score matching. Some other researchers use rich sets of covariates and the learned structure from neural networks to predict outcomes for counterfactual test cases and estimate causal impacts at the individual level (Hassanpour and Greiner, 2019; Johansson et al., 2016; Louizos et al., 2017; Shalit et al., 2017). These deep matching approaches are applied to the data from well-known RCTs, and their average treatment effects are generally close to those obtained from the randomized experiments.

As deep learning researchers aim to improve models’ generalizability across distributions and populations, data is increasingly becoming at least as important as methodology. In the controlled setting of the game of Go, the use of challenging self-generated training cases dramatically improved model performance (Silver et al., 2016, 2017). As the lion in the city example illustrates, confounding variables can be deep and difficult to learn, and thinking about counterfactuals has the potential to identify models’ blind spots. The adversarial examples that researchers often create through slight perturbations to training data help to improve models’ robustness, in part by simulating the curiosity-driven learning of biological systems that Sinz et al. (2019) describes. Augmenting training data with more severe and stranger modifications has the potential to do even more.

### 3.3. Interaction Effects

Given the many constraints associated with running large RCTs or finding credible quasi-experiments, the extent to which an estimated causal impact can generalize across populations has long been a focus of the statistics literature and is known to statisticians as external validity (Campbell and Stanley, 1963; Cook and Campbell, 1979; Imbens and Angrist, 1994; Card, 2001; Heckman and Vytlacil, 2005). It is the need to capture cross-subject variation in impacts—and to better generalize to other populations—that has led to increased interest in interaction effects and the learning-based models that can capture them.

One of deep learning’s key contributions to measurement and inference is the ability to identify complex and deep features that arise through multilayered interactions across many input variables. Future research could apply neural networks to existing data from a wide range of RCTs and natural experiments to shed light on how those impacts vary across subjects and how the effects can be effectively generalized to other populations. Easy-to-use tools that enable applied
researchers to leverage network-based approaches could be particularly valuable.

This importance of interactions in generalization is paradoxical. The bias-variance tradeoff shown in Figure 2 suggests that the ability to generalize should increase as parameters are subtracted from the model. Taking account of additional interaction effects across predictors would push model complexity in the opposite direction. Estimating deep interactions requires large numbers of parameters.

Findings from the neuroscience literature may help to resolve this apparent contradiction. In reviewing the determinants of mental and behavioral flexibility in *Drosophila*, Devineni and Scaplen (2022) describe a particular interaction effect that provides a helpful motivating example. Fruit flies have an innate aversion to the smell of carbon dioxide, which is a warning signal emitted by other flies. In situations of extreme hunger, however, the brain releases chemical signals that cause this response to shift. Flies seek out the odor, because it raises a possibility of food that outweighs the risk. Hence, the modulation of priorities across different contexts is handled by a system that operates at a global level and is distinct from the mechanisms that process innate and learned responses. In this case, the modulation is straightforward. Food has become a higher priority, and the chemical signals tilt decision-making to reflect this preference.

While the effects of context are not always so straightforward, this example from fruit flies highlights one way in which priorities can shift in response to context while keeping the essential structure of a decision-making process intact. Along similar lines, deep learning models could incorporate dramatic changes in the population of test cases through relatively simple responses. For instance, hyperparameters such as activation thresholds could be designed to depend upon case- or distribution-specific factors.

4. Domain Generalization

In some situations, the nature of test cases encountered—or the behavior of the target function in response to those cases—differs so much from that experienced in training that the model must be modified in order to accommodate it. When generalizing from one distribution to another, the focus is on developing flexible models that can perform well across the diversity of cases observed in the population. When the changes are sufficiently extreme, however, it is no longer helpful to think about one large model that does everything. What we call generalization happens at a higher level of abstraction, where there is different version of the model for each domain.

The types of changes in domain that researchers consider can be divided into two broad categories. In the first, domain varies continuously, sometimes in ways that are not immediately apparent. The second form of domain change involves a shift to a distinct set of cases in a new environment.

4.1. Concept Drift

In the gradual form of domain change known as “concept drift,” it is not only the distribution of covariates that shifts, but there is some change in the ground truth target function mapping those covariates to labels. Examples of this change include the distortion of images or sounds and the evolution of words’ meanings in texts. Surveys of this form of domain change and strategies to address it can be found in Bayram et al. (2022); Lu et al. (2019a,b); Wang et al. (2018), and Yang et al. (2021).

A well-known early example of concept drift is Google Flu Trends. The project, first announced in 2008, produced a highly effective predictor of flu patterns in the United States based upon Google users’ search activity. The performance of the model declined, however, in successive flu seasons as search activity changed, partly due to the rollout of a new “autocomplete” feature in Google Search that changed the likelihoods of different searches (Lazer et al., 2014).

Henriksson et al. (2021) note that, for some applications, performance degradation of this form poses safety concerns. The authors test a framework in which a supervisor monitors the classification probabilities generated by a trained neural network and alerts the user if they indicate an increase in the rate of outliers or a change in the distribution of test cases being evaluated. The authors find that the supervisor’s ability to identify distributional changes generally improves with the training accuracy of the model but is inconsistent, varying widely across training epochs. Similarly, Rohlfis (2023b) finds among a sampling of neural networks for object recognition that the models’ self-assessments of classification confidence are effective proxies for accuracy among in-distribution test cases. When given low-resolution variants of those images, however, the classifiers’ performance declines, and the models are overconfident in their classifications.

Seen from the lens of the bias-variance tradeoff in Subsection 2.1, this phenomenon through which the target function changes but parameters are fixed is a form of underfitting. Ants and birds whose patterns of behavior are locked down by evolution are insensitive to data they receive about their situation. Similarly, the model’s training parameters have been locked down. The model’s forecasts are still responsive to changes in the features associated with new test cases. It is not just the covariates that are changing, however. Like in the Google Flu case, the target function that maps covariates to output classes is changing. Because the model’s parameters do not change, the model is unresponsive and therefore underfit with respect to this change in the target function.

In shallow learning contexts, Wang et al. (2018) find that, when domain change occurs in a continuous fashion, much of the degradation in performance reflects changes in the likelihoods of different classes and can effectively be addressed by monitoring those probabilities and adjusting the corresponding thresholds. Threshold-based adjustment

---

5 As many authors note, varying definitions of concept drift are used in the literature. For the purposes of this discussion, a definition of gradual and potentially imperceptible domain change will be used.
Generalization in neural networks

Figure 5: Concept Drift

of this form is not unlike the adjustment seen in fruit flies, discussed earlier, in which starvation causes the insect to modify its behaviors—for instance, the amount of stimulus required for a given action—to increase the likelihood of obtaining food.

The phenomenon of changing thresholds is illustrated in Figure 5, which is adapted from Bayram et al. (2022). They provide a useful review of studies of concept drift and its varying definitions in the literature. In the example, there is some boundary between the green circles and the violet triangles that was learned from training data. Drift occurs in that, among new test cases from the triangle class, the values of the second predictor $X_2$ tend to be lower than seen in training. In some cases of concept drift, researchers aim to determine when a substantive change in domain has occurred so that the model should be reconfigured before further use. Other drift-based approaches aim to develop classifiers that are robust to such continuous domain variation. Bayram et al. (2022) note that, when attempting to measure whether drift has occurred, performance-based tests of changes in distribution and domain have a tendency to raise false alarms, because performance may degrade for reasons such as data quality that would not be solved through reconfigurations to the model.¹

Xiang et al. (2023) and Yuan et al. (2022) provide useful surveys of strategies through which neural networks have been adapted to accommodate changes in concept definitions. In time-series contexts, some researchers partially or fully refit model parameters on a regular basis. This process of putting greater weight on recent observations and sometimes discarding earlier cases helps to prevent learned systems from using stale concepts. As Yuan et al. (2022) also note, researchers have also found ways to adjust structural elements to accommodate concept drift, adjusting the width Kauschke et al. (2019) and depth Pratama et al. (2019) in response to rising rates of misclassification.

In visual object-recognition problems, rotated or distorted images are examples of concept drift. The changes in features could be subtle enough that they do not impact the labels that a human would apply to the cases—so that the true input-output relationship would be unaffected. As with generalization across distributions, however, a classifier trained on undistorted data would have difficulty accurately classifying the new cases.

Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) problems constitute one everyday setting to which researchers have applied tools to recognize and adapt to concept drift. These tests check to ensure that a computer user is human by presenting the user with a text or image recognition problem that would cause difficulties for an automated system. CAPTCHA tests that are used to secure computer systems against automated attacks constitute a compelling category of problems—and one that has been studied extensively—in which context and feature distributions vary continuously. Many CAPTCHAs rely upon the recognition of abstract features—e.g., distinguishing letters that are deformed, overlapping, or have distracting elements such as noise or ancillary lines. A set of characters written in this deformed and overlapping style is illustrated in Figure 6. The image is taken from a CAPTCHA-inspired font by Rogers (2014). Optical character recognition (OCR) approaches typically distinguish text in a sequential manner, first determining boundaries to isolate individual characters and then determining which letter or number each of those characters represents. While it does not cause great difficulty for humans, the overlapping of characters in the image confounds the automated approach.

The wide variety of CAPTCHA-style problems available provides the deep learning community with an extensive amount of human-labeled data to illustrate what features define different classes of objects and what features can be altered without causing their labels to change. Xu et al.

¹In a related survey, Yang et al. (2021) discuss the problem of detecting if a case falls outside of the trained distribution. Similarly to Bayram et al. (2022), the authors find a diversity of terminology and propose a unified framework for considering such situations.
Table 4
Datasets commonly used for Learning Domain Generalization for Classification Tasks

| Medium | Dataset | Classes | Domains |
|--------|---------|---------|---------|
| Images | VisDA2017 (Peng et al., 2017) | Plane, bike, bird, boat, wine, bus, car, dog, horse, motorcycle, person, plant, skateboard, train, truck | Synthetic and real-world images |
| Images | Office-Home (Venkateswara et al., 2017) | 65 classes of everyday items | Art, clipart, product, and real world |
| Images | Office-31 (Saenko et al., 2010) | 31 classes of office objects | Amazon product pictures, Webcam photos, DLSR photos |
| Images | Caltech Camera Traps (Beery et al., 2018) | 17 types of animals, car, or no object | 140 fixed camera locations |
| Texts | Reuters-21578 (Lewis, 1997) | Newswire article subjects | Categories of article subjects: Exchanges, Organizations, People, Places, Other |
| Texts | Enron Emails (Klimt and Yang, 2004) | User-defined folders | Email users |
| Texts | Multi-domain Sentiment Dataset (Blitzer et al., 2007) | Amazon product ratings based on review texts | Books, Kitchen, Electronics, and DVDs |

4.2. Observable Shift

In the second form of domain change, when there is a dramatic and observable shift, many features of the input data distribution as well as the target function are dramatically altered. Sufficiently many key elements of the input-output relationship are retained, however, for generalization from the old domain to be possible. Examples could include a change in the identity of the speaker for a speech recognition model or the application of a text processing tool to a foreign language. Importantly, the researcher does not have access to labeled training data for the new domain. To classify the new test cases, it is therefore necessary for the model to learn how to generalize across domains.

Figure 7, reproduced from Xu et al. (2019), illustrates one compelling dataset that researchers have used to train neural networks to perform this type of generalization. The images in the PACS data represent objects from seven classes including horse, person, house, guitar, and giraffe, as depicted in the figure, as well as dog and elephant. The four rows of images illustrate the four domains in which these different classes of objects are represented: art paintings, cartoons, photos, and sketches. As the example images illustrate, each class has some distinctive traits that appear with some consistency across domains, such as the spots and neck on a giraffe and the A-shaped roof of a house. The exact ways in which those traits appear on the images varies considerably, however, across the domains. Each domain has its own complex, nonlinear relationship through which features map to these seven classes. For the purposes of learning domain generalization, a neural network is trained to identify these different object types using data from three of the four domains, so that it has experience not only with the classes but in how their definitions tend to vary from one domain to the next. It is then tested on its ability to perform the same task in the new domain.

Table 4 describes a sampling of this variety of datasets used for developing domain generalizing models. The four columns describe the input type (images or texts), the dataset name and source, the classes that are learned within a domain, and the different domains in which these classes are defined. The first eight datasets describe object identification problems that are similar in structure to that posed in PACS. The numbers of classes range from five to 345. Some of the datasets like PACS have diverse sets of domains...
to learn, while in other cases, the domains are relatively similar, consisting of photographs taken from different image libraries. The text-based datasets present tasks of identifying the subjects of newswire articles (with domains consisting of different broad categories of subjects), sorting emails into folders for different users, and projecting the consumer rating based upon the review text for different types of Amazon products.

4.2.1. Transfer Learning
One popular and effective strategy for developing models to generalize across discrete changes in domain is transfer learning. Through this approach, a neural network is designed to adapt knowledge acquired in one domain and apply it to another. Table 5 summarizes the variety of transfer learning strategies that have been used in the literature. For greater detail on these approaches and the variety of problems to which they have been applied, Gulrajani and Lopez-Paz (2020); Niu et al. (2020); Wang and Deng (2021); Wang et al. (2023); Zhou et al. (2022), and Zhuang et al. (2021) provide extensive and useful reviews.7

Transferring network-based learning across distinct domains takes one of two conceptual approaches. In transductive domain adaptation, there is explicit translation from one domain to another. Such strategies require some knowledge of the new domain at the time that the model is developed. Typically, there is a dataset of unlabeled cases from the new domain that is used to determine how the patterns of features differ between the old and the new setting. Inductive models, on the other hand, are developed for use in any arbitrary domain. The researcher supposes that there is a core, domain-invariant model that represents the archetypal versions of the classes being modeled. Domain-specific training data are used to learn both about the particular patterns exhibited in that domain and also about the core model. Inductive transfer learning can use unlabeled feature data from the new domain during model development, but it is not required. Gulrajani and Lopez-Paz (2020) provide a brief survey of approaches and datasets for this form of domain generalization, such as those used by the original Li et al. (2017) on PACS, and the authors provide a set of associated algorithms and datasets in their DomainBed package for use with PyTorch (Paszke et al., 2019).

The first of the strategies from Table 5, feature mapping, can transductive or inductive. An inductive variant is illustrated in Figure 8a. In feature mapping approaches, there is a learned transformation function that processes input features from a given domain so that they can be consumed by a classifier from another domain or a domain invariant classifier, as in the diagram. The top of the figure shows pictures of elephants from three domains:
cartoons, sketches, and photographs, all from the PACS data. Each domain has its own feature transformation function that converts the input into the format expected by the classifier. The transformation often expands the input space, so that the generalized representation shown in the diagram is not necessarily viewable as an image. All domain-specific computation is performed on the features, and the remaining components of the classifier are shared across two or more domains. In some cases, researchers use feature transformation to supplement the training process for the domain-specific classifiers. Shankar et al. (2018), for instance, employ generative adversarial learning in which transformed cases from other domains are used to create adversarial examples to refine the classification and domain-identification components of the network.

The next domain generalization strategy listed in Table 5, parameter sharing, is illustrated in Figure 8b. As with Figure 8a, the example presented in the graph involves inductive transfer. As Figure 8b illustrates, the parameter sharing approach involves a separate network-based classifier for each domain. Optimization proceeds in a simultaneous fashion, with restrictions applied to the parameters to ensure that the domain-specific classifiers are sufficiently similar to the domain invariant one. This approach can also be applied transductively, in which case there is no domain invariant model, but parameter restrictions are applied to maintain similarity between the domain-specific models.

In the final generalization approach from Table 5, ensemble weighting, multiple neural networks are developed during the training process: one for each of the domains that is available during training, with no domain invariant model. When test cases are observed from the unseen domain, they are run through each of these trained models, some decision rule such as majority rule or averaging of probabilities is used to aggregate these domain-specific projections into a final classification.

### 4.2.2. Empirical Results

Next, we examine how some of these strategies for generalization have performed in the domain generalization exercise from the PACS data. Table 6 lists the names, attributes, and performance statistics for several recent models together with the original PACS study from Li et al. (2017). The best-performing specification is presented for each model, with the model names, specification details, and performance statistics matching those presented in the website Papers With Code (2024). Each specification performs four domain generalization exercises in which one of art painting, cartoon, photo, or sketch is the unseen target domain, and the other three are the source domains. Because there are multiple source domains, each of the specifications here uses inductive transfer and a domain-invariant classifier from which to generalize to the unseen domain. For all of the models, the average accuracy across the four unseen domains is shown in the table. For papers that report the results of the four separate generalization exercises separately, those breakouts are also shown. The two separate entries for the UniDG model include the best-performing specification overall as well as the best-performing specification for which this breakout is reported.

Each of these specifications shown in Table 6 has a model name as well as a “backbone” presented. The backbone refers to an established foundation model whose architecture and trained parameters are used as a starting point. Some of these models, including AlexNet Krizhevsky (2014), ConvNeXt Liu et al. (2022b), RegNetY-16GF Radosavovic et al. (2020), and ResNet-50 He et al. (2016) are trained using ImageNet-V2 (Russakovsky et al., 2015), the popular object-recognition data discussed in Subsection 2.4. Those for which “Multiple” is listed as a backbone use more than one foundational model built off of those data. For some of those backbones, the Stochastic Weight Averaging Densely (SWAD) regularization technique from Cha et al. (2021) is used. Some other of these models use as a foundation model the multimodal Contrastive Language-Image Pre-Training (CLIP) approach of Radford et al. (2021) and its OpenCLIP
variations. Both supplement image data with text in order to improve classification. The underlying network architectures used for these models are variants of the Vision Transformer (ViT) set of models from Dosovitskiy et al. (2020).

In addition to backbone, a “strategy type” is listed for each model. While each study introduces its own model-specific innovations, this field in the table indicates which of the general categories of transfer learning approaches from Table 5 each one use. UniDG is labeled “Feature and Parameter” as that model combines elements of both feature mapping and parameter sharing. Due to the variety of foundational models and datasets used, the results presented in the table should not be construed as a direct comparison of the effectiveness of different generalization strategies. They should instead be regarded as an illustration of the different approaches used and how improvements in many dimensions have enhanced neural networks’ ability to generalize across domains.

As the last column from Table 6 indicates, the ability to generalize across visual domains has improved considerably over the time that researchers have been working on the problem. While the initial Li et al. (2017) study only achieved 69.2% accuracy on average, recent approaches have reached as high as 99.0%. Among the three categories of approaches, feature mapping is the most commonly represented among the high-performing models, but strong performance can be seen from parameter sharing and ensemble weighting as well.

For the models that report breakdowns for their results, generalization is the consistently the strongest when “photo” is the unseen domain. This finding is relatively unsurprising, given that the foundation models are built on photographic data. Performance tends to be worst on the “sketch” category, whose images tend to be the most abstract and to provide the least visual information.

5. Task Generalization

Moving further to the right in the specificity-generality continuum in Figure 1, the next higher level of abstraction is generalization across tasks. The distributions and domains may still be changing, but that is not necessary. The defining element of this form of generalization is the output classes or specific question being answered is different from in the original model. It is thus absolutely necessary to reconfigure the model, because the format of the outputs has changed.

This section begins with a discussion of early work to train a neural network to perform a new set of skills while retaining previously acquired knowledge. It then discusses advances in the use of backbone models that, having learned how to perform a general set of tasks, can learn new variations with a small number of training examples, a process known as few-shot learning. We then explore recent innovations in text-based deep learning, where the transformer architecture, another backbone-driven strategy, has helped researchers to produce language-based knowledge that is highly portable across tasks. While few-shot learning only expanded the set of outcome classes, these language tasks are diverse, including different types of questions and output formats. We then discuss how this growing ability to generalize knowledge across tasks has led to the rise of widely-used transformer-based foundation models, with increased consolidation and centralization of network-based learning for core functions.

5.1. Catastrophic Forgetting

In biological systems, learning is continuous and online, so that encountering new data can lead decision rules to change (Sinz et al., 2019). At the same time, stale information is steadily discarded through the pruning of unused synapses, but it is done slowly to maintain stability and manage risk (Akers et al., 2014; Richards and Frankland, 2017). Researchers working with artificial systems face this same challenge of balancing new and existing skills.

The most common form of task generalization considered in the literature is to train a classifier to identify one or more new categories. Early work approached the problem from the perspective of expanding the possible outcomes for an existing trained classifier. Researchers observed a phenomenon of catastrophic forgetting, whereby acquiring a new skill reduced the system’s ability at previously learned tasks. Vandenhende et al. (2021) and Zhang and Yang (2022) survey a variety of approaches for adapting networks’ structures and other design elements to handle multiple tasks without such difficulties. A few are discussed here.

Strategies to avoid catastrophic forgetting have mainly involved modifications to the optimization problem used to fit the parameters of the neural network, including Progressive Learning (PL), elastic weight consolidation (EWC), and synaptic intelligence (SI), all of which have been effective in some applications but come with costs. In PL, a subset of the learned architecture from prior cases is maintained and repeatedly consulted for comparison when new cases are encountered (Rusu et al., 2016b,a). This approach has proven to be effective in the completion of mazes and video game tasks. It lacks scalability, however, as it expands the size of the network and the amount of time and resources required with each new training observation. Fayek et al. (2020) partially address this limitation with a pruning procedure that regularly removes the previously trained weights whose estimated magnitudes fall below some threshold.

EWC and SI both vary the speed at which parameter updates occur depending upon their relevance to previously learned tasks. EWC does so through Bayesian updating with task-specific constraints added at a global level to the optimization problem to account for different parameters’ impacts on performance in other tasks. A more localized modification of the loss function is performed in SI. Associated with each connection is both a weight and a degree of importance of that parameter in previously learned tasks, where higher importance increases the cost of updating the weight. Researchers have found both approaches to reduce the problem of catastrophic forgetting while maintaining high performance at multiple object identification tasks or
video games (Kirkpatrick et al., 2017; Liu et al., 2020; Zenke et al., 2017). Like PL, however, the strategies introduce new problems related to cost and scalability. Under both EWC and SI, as the system learns additional tasks, it becomes increasingly constrained and slow to learn (Schwarz et al., 2018). All three methodologies address the problem of forgetting skills, and work along these lines has contributed to progress in knowledge distillation (Gou et al., 2021), but the networks are difficult to scale due to their continually growing demands for computation and storage.

Some recent progress toward stabilizing deep learning networks aims to maintain scalability by addressing catastrophic forgetting without repeatedly reviewing training cases. A biologically-inspired approach by Pandit and Kudithipudi (2020) is patterned after the “neurogenesis” that regulates memory and forgetting in animal brains (Akers et al., 2014). The approach, like SI, introduces local modifications to the network’s loss function and updating rules. The neural network that the authors design handles memory and storage through the gradual replication of commonly used neurons and connections and the termination of unnecessary ones. In a Bayesian-style approach, Teh et al. (2017) train a modular RL network whose components are coordinated by a central policy. Worker modules are assigned distinct tasks and solve their optimization problems separately, but each acts in reference to the common policy and considers closeness to this policy as one of its objectives. Each task’s optimal solution in turn contributes to the determination of the general policy. The authors test this approach on multi-room maze and laser tag problems and find that it performs effectively and that its solution are robust to small and moderate-sized changes in hyperparameter values.

As Vandenhende et al. (2021) and Zhang and Yang (2022) discuss, some fruitful strategies for generalizing across tasks employ approaches similar to the policy coordination approach of Teh et al. (2017) through the use of transfer learning. In a feature-mapping variant, task-specific models dedicated to the same task share a common core of network layers that serve as a “trunk,” potentially separate feature transformers (if their inputs vary), and task-specific output layers or “heads.” In cases of computer vision, separate heads might exist for object detection, semantic segmentation of images, and object recognition. Alternate parameter-sharing transfer-based strategies allow each task-specific model to have its own trunk but with constraints that maintain similarity across the heads. These transfer approaches can be seen as variations on the transfer strategies described in Figure 8 for domain generalization, but with distinct output heads for different tasks. The growth of transfer-based strategies such as these, based upon foundation models as backbones, has emerged as a key means through which existing models can be trained to perform new tasks without degrading earlier performance. The next two subsections examine powerful variations on this approach—first to recognize objects with very few examples and then to learn varied natural language tasks.

5.2. Few-Shot Learning

Exciting progress has been seen recently in a particular form of task generalization known as few-shot learning, a setup that is illustrated in Figure 9. The images in the figure are taken from the mini-ImageNet dataset and the structure of the two problems aligns with tests performed by Vinyals et al. (2016). The network has access to a large number of training cases of different types. Examples are presented in Figure 9a. The images shown in the figure are all from different classes, but the data include multiple examples within each category. During the training period, there is a holdout set of classes, and the model is not allowed to view any of the cases from those classes. In testing, the model is given a set of new classes to distinguish. Figure 9b shows an example of a relatively easy five-way discrimination task in which the five classes are chosen from the dataset at large. The model is exposed to a small number of examples from each class (two in the example) through a process described as tuning, and it is then evaluated in testing on its ability to distinguish images from the five categories. Figure 9c shows a more difficult task in which all five classes are descended from the broader set of dog breeds, so that object identification requires a finer level of distinction.

Some approaches to few-shot learning transfer skills from networks trained to perform more traditional image classification problems. (Vinyals et al., 2016), for instance,
learn feature embeddings from ImageNet to enable distance-type comparisons of test versus support images. In these models, much of the network-based learning is of embeddings to represent the features from individual images. Discrimination across classes is performed by applying a metric with which to compute the distances between each of the test images and the support images. Features of the distance metric may be learned in the training process, but the training images do not include elements from the support classes considered in testing. This distance-to-embedding approach is illustrated in Figure 10a. A common network is used to transform image features into the embeddings consumed by a pairwise distance function which is used to evaluate image similarity between support and test cases.

Another popular strategy of meta-learning, as in the Model-Agnostic Meta-Learning (MAML) approach of Finn et al. (2017), which builds on the approach of Andrychowicz et al. (2016), replicates in training the tuning and evaluation processes performed in testing using a variety of support and test sets constructed from a training set of images. What is learned by the network is a generalization approach, as illustrated by the pink arrow. That approach itself consumes labeled support images as inputs and produces labels for test images as outputs. Competing generalization approaches are evaluated on the accuracy of their label assignments for the test cases. In testing and deployment, labeled support images and unlabeled test images are supplied as inputs, and the output is a mapping that labels the test images. As Figure 10b illustrates, training cases for this model consist of support-test combinations, and the output is a learner that can be tuned on new support cases. Approaches to few-shot learning are described in greater detail in surveys by Li et al. (2021); Wang et al. (2019c), and meta-learning is reviewed more broadly in Hospedales et al. (2022).

The levels of performance of three state-of-the-art few-shot learners are presented in Table 7. Two common datasets are used: the CIFAR-FS (Bertinetto et al., 2019) and Mini-ImageNet (Vinyals et al., 2016). Both have structures similar to that illustrated in Figure 9, and performance is evaluated here on the easy type of problem of discriminating across randomly selected holdout classes. These datasets are subsets of the CIFAR-100 (Krizhevsky, 2009) and ImageNet ILSVRC-2012 (Russakovsky et al., 2015) sets of images that have been configured for few-shot learning. Results are shown in all cases for 5-way few-shot estimation. Performance is evaluated by introducing the model to five new classes of images, giving it one or five examples of each, and testing its ability to accurately discriminate among test images from those five classes. The P>M>F model of Hu et al. (2022a) involves three steps of pre-training an image classification model, meta-learning the task generalization process, and fine-tuning to the support data for testing. Its pretraining employs as a backbone the DINO (Caron et al., 2021) self-supervised variant of the ViT (Dosovitskiy et al., 2020) model, and it uses the ProtoNet algorithm (Snell et al., 2017) for its meta-learning step. Like Vinyals et al. (2016), the Self-Optimal Transport (SOT) approach of Shalam and Korman (2022) involves the learning of feature embeddings for distance-based comparisons, using the ProtoNet algorithm (Snell et al., 2017) on a small, customized convolutional net with a ResNet-12 backbone for fine-tuning (He et al., 2016). The PEMnE-BMS strategy of Hu et al. (2022b) supplements feature transformation-based transfer learning with additional steps including boosting, with a Wide ResNet (WRN) (Zagoruyko and Komodakis, 2017) backbone following the specification of Mangla et al. (2020). As the table shows, these recent top-performing models execute this task with an rate of accuracy of 84.3%
to 95.3% for one-shot and 88.8% to 98.4% for five-shot learning.

5.3. Transformers

An additional area of task generalization in which exciting progress has been made in recent years is the transformer architecture introduced and popularized by Vaswani et al. (2017). Transformers build upon the autoencoder architecture for capturing deep features to reduce the dimensionality of complex data, popularized by Hinton and Salakhutdinov (2006) and Masci et al. (2011). The transformer architecture builds upon this encoding of deep structures by increasing the richness of the relationships among them. Like Recurrent Neural Networks (RNNs), transformers operate on sequential data like words in a sentence. Importantly, however, the transformer structure’s “self-attention” mechanism allows for higher levels of connectedness and dependency across elements that are far away from one another than was achieved through RNNs.

The transformer architecture has proven to be highly effective at improving model performance in a variety of tasks, particularly those related to vision and language. Chaudhari et al. (2021) reviews the architecture and a variety of applications. Key innovations in vision are discussed in Dosovitskiy et al. (2020) and a survey by Khan et al. (2022).

Transformers’ ability to understand relationships among distant elements of a sequence has proven to be critical in the processing of natural language data, in which a word like “this” or “he” can reference an entity mentioned far earlier or later in a sentence or phrase. One highly influential application of transformers in the NLP setting is the Bidirectional Encoder Representations from Transformers (BERT) approach for language comprehension introduced by Devlin et al. (2019).

| Model               | Test               | Performance |
|---------------------|--------------------|-------------|
| Human Benchmark     | GLUE (9 tests)     | 80%         |
| DeBERTa (He et al., 2021) | (Wang et al., 2019b) | 90.0%       |
| ALBERT (Lan et al., 2020) |                     | 89.4%       |
| Human Benchmark     | SuperGLUE (8 tests) | 89.8%       |
| DeBERTa (He et al., 2021) | (Wang et al., 2019a) | 90.3%       |

The overall structure of the BERT model is depicted in Figure 11. Figure 11a illustrates the training process. The inputs are pairs of potentially related phrases with some words redacted. In the example, they are the first two sentences of the Gettysburg Address (Lincoln, 1863). The features from these sentences are transformed into vector representations. For each word, these embeddings capture context by characterizing distributions of words that tend to appear close to that word. As “bidirectional” in the model name implies, such nearby words are identified through forward and backward traversal of the phrase and consequently include words appearing both before and after the word being described. The embeddings are subsequently fed as inputs into two classifiers. The first determines the identity of the masked words. The second determines if the two sentences are related in the sense of occurring one after the other in a text. The entire process is trained simultaneously.

Tuning and deployment of the BERT model is shown in Figure 11b. In testing and deployment, the embeddings, the distributions of likely words, and the classifications of sentence relatedness are combined into a broader set of embeddings that describe pairs of phrases. Subsets of those embeddings are used as inputs for task-specific classifiers.
The training of those task-specific classifiers is described as tuning or fine-tuning. The model itself becomes a feature transformer, consuming phrases or pairs of phrases and outputting the intermediate embeddings, the probabilities for the masked words, and the sentence similarity measure. Subsets of this complete set of embeddings are used as inputs for specific applications such as sentiment analysis or rating the correctness of potential answers to supplied questions. Each of the applications is trained separately through a process described as fine-tuning.

Table 8 presents the results of two recent BERT-based networks, DeBERTa and ALBERT, on two batteries of phrase comprehension tests that are commonly used to evaluate NLP engines. The General Language Understanding Evaluation (GLUE) and SuperGLUE measures are averages across batteries of nine and eight tests, respectively, and NLP engines are assessed on their average performance. Typical human performance is provided as a benchmark. As the table illustrates, for both tests, state-of-the-art BERT-based methods outperform humans.

### 5.3.1. Transformers as Foundation Models

Because transformer-based architectures are so versatile and portable, researchers are increasingly relying upon pre-trained foundation models. The use of such models mirrors the strategies used in domain generalization and few-shot learning, through which a model trained on a large dataset can then be applied to slightly different problems with relatively little domain- or task-specific training. Established neural networks trained on ImageNet, such as those described in Subsection 2.4 on sample generalization, are examples of foundation models. The popularity of transformers has led to the application of this strategy on a larger scale. Popular toolkits for using foundation models include Wu et al. (2019) in the vision setting and Wolf et al. (2020) for NLP. Bommasani et al. (2021) discuss the patterns and implications of this industry-wide shift.

Table 9 summarizes some transformer-based foundation models that are popular now and some tasks to which they are commonly applied. In addition to those dedicated to vision-related tasks and those specific to NLP, some use data from both modalities to inform one another, a form of generalization that we discuss in greater detail in the next section.

### 6. Modality Generalization

Artificial neural networks are typically specific to a datatype, considering only images, only text, or only audio, for instance. Evidence from the neuroscience literature suggests that, by contrast, some important elements of learning and memory operate at a more abstract level in which information across different sensory sources is consolidated. In conditioning experiments on fruit flies, for instance, Ueno et al. (2017) finds that some stimuli only lead to learned responses when researcher-generated odors and lights are simultaneously present. The authors propose that this phenomenon may arise due to stability-enhancing dual-gated learning pathways. Additionally, evidence from Quiroga et al. (2005) suggests that human brains may have cells dedicated to specific individuals like celebrities, so that seeing a picture of American actress Jennifer Aniston or just her name would activate the same location in the brain.

In the same way that biological brains abstract learn and remember entities and phenomena in a way that is distinct from the information source, some research in artificial neural networks transfers learning across modalities—that is, across different data formats or spheres of knowledge from those of the original model. It is a broader form of generalization than distribution, domain, and task and is in a sense a drastic change in all of these at once. The literature in this area is not extensive, but it is growing. Some recent influential transformer-based models, for instance, achieve high levels of performance at vision-related tasks by exploiting deep and extensive cross-model information. The CLIP model (Radford et al., 2021), used as a backbone in some of the domain generalization models described in Subsection 4.2.1, is one example. Such models incorporate information on how different types of images have been described with text as well as learned skills on how text-based concepts relate to one another. This combined sourcing of datatypes enables the model to make connections and identify related concepts more effectively than is achieved with visual data alone.

Table 10 summarizes the findings from three innovative studies that help to illustrate possibilities in this area. The first two lines of Panel A present results from the Deep
Transfer Networks (DTN) approach of Shu et al. (2015), which builds upon a related non-network based classification approach by Qi et al. (2011). In that model, a feature transformation function is trained based upon web-based images together with accompanying text descriptions. A classifier is then trained to label the images based upon these text-enriched embeddings of the image data. In deployment, the transformer generates embeddings for images that do not contain text descriptions—essentially describing text that tends to accompany images similar to the current input—and those embeddings are fed as inputs into the classifier. While newer transformer-based approaches improve upon the text-to-image generalization ability, this study is instructive because of its comparison of performance with and without the supplemental text-based data. The first line of the table presents the results of a baseline classifier using a Stacked Auto-Encoder that only includes data on images. The second line shows the classification results for the DTN model that incorporates the feature transformer trained with text descriptions. Results are shown from the authors’ tests that include 10 training images per class. As the results show, the approach using embeddings substantially improves classification performance over the benchmark.9

The next two lines of panel A present results from Vasu et al. (2024). That study compares the ability to classify ImageNet images between two versions of CLIP (Radford et al., 2021) using different levels of data quality. The baseline version uses web-scraped data from OpenAI with imprecise and coarsely-written captions. The enhanced dataset used by the authors is synthetic and more descriptive captions that were machine-generated separately for each image using a variation of the captioning model by Yu et al. (2022) and developed as part of the DataComp competition for improved CLIP performance (Gadre et al., 2023).10 As the results show, the use of greater detail and precision in the captions—even the synthetically generated kind that themselves derive from language-based transformer models—lead to a noticeable improvement in image classification performance.

Panel B of Table 10 describes a modality generalization approach that is inspired by the structure of animalian brains. Borthakur and Cleland (2019) begin with a biologically-inspired classifier with numbers of nodes, levels of connectivity, and other design features based upon that of the mammalian olfactory system. They then examine the extent to which a set of hyperparameters developed for one problem of classifying odorants can be applied to different problems with comparable effectiveness. They develop their Spiking Neural Network (SNN) to classify odorants from gas sensor data in an online learning context, and they then re-train the classifier on data from the same sensor after it has aged three years. In the language of Figure 1, this first test can be viewed as something between distribution and domain generalization, as in the case of concept drift. The authors then apply the same network structure with the same hyperparameters to the classification of different types of Japanese forests from satellite images and to the classification of frog and toad species based upon auditory information. They find that their SNN setup retains a high degree of accuracy in these widely different problems. While changing the context and problem degrades the performance somewhat, they find that similar accuracy as the original classifier can be achieved without hyperparameter re-tuning by increasing the number of training cases.

7. Scope Generalization

In the mid-20th century, much of the work on artificial intelligence (AI) focused on codifying and automating logical reasoning. Early advances in this area, known as symbolic AI and referred to by Haugeland (1989) as “good old-fashioned
AI, “engendered a sense of optimism with bold projections of the rate at which the field would develop Simon and Newell (1958). The decades that followed, however, are widely regarded as an “AI winter” of disappointing progress. When, in the late 20th and early 21st centuries, rapid advances in AI began again to surpass expectations, it was through the emergence of the powerful but less transparent technologies of machine learning and artificial neural networks (Russell and Norvig, 2020).

In the process of exploring and refining these newer approaches, some research has begun to revisit AI’s origins in symbolic reasoning. Generalization across scopes—that is, generalization at the level of abstract principles—and the field of neurosymbolic AI involve two ways of translating between human and machine understanding. The first, knowledge representation, searches for ways in which logic and human insights can be incorporated into deep learning strategies to enhance their performance. The second, knowledge extraction, seeks to translate the outputs and reasoning from artificial systems so that they can be better understood by humans.

7.1. Knowledge Representation

Some of the greatest challenges for modern AI involve the distillation of facts into principles and the application of those principles to new areas of knowledge. Davis and Marcus (2015) highlight the amount of semantic knowledge that humans regard as commonsense by considering the picture shown in Figure 12a of Julia Child’s kitchen.11 A human viewer would naturally infer that four legs are holding up the tabletop and some sort of nail, hook, or adhesive is keeping the pictures on the cabinet doors—and that without them, they would all fall to the floor due to the force of gravity. While these thoughts might not occur to someone consciously, violations of such principles would be immediately noticeable. This example of the kitchen photograph helps to illustrate the many inferences that humans draw from stimuli and the depth of understanding required to replicate that process artificially.

Biological systems process information in a variety of ways. Some components of brains perform pattern recognition like deep learning does. Other less-understood components perform logical reasoning. No data-driven machine-based system yet exists that learns logical and symbolic principles effectively from scratch. It is not clear how such a learning system would be designed or what data it would use. In the meantime, however, researchers have pursued a hybrid approach in which humans compile their own learned principles and logical relationships and then train networks to apply them.

Substantial progress has been made to establish a base of facts and relationships among them as well as the tools necessary to expand upon that set of information. As Bader and Hitzler (2005), Besold et al. (2017), and Townsend et al. (2020) note in surveys on neurosymbolic AI, some research has worked to incorporate human sense and domain expertise into deep learning systems. In some neural-symbolic cycles, the machine could be used to identify objects in an image, as in Figure 12b, in which objects have been identified using the Detectron2 package from Meta (Wu et al., 2019).12 Researchers could include additional “neural gates” into the architecture that would capture logical conclusions. For instance, a human-designed gate might fire only if more than one cooking appliance is identified in a photo. Such logical components could help artificial systems to more quickly recognize that the room being pictured is a kitchen. This symbolic logic that would be used for deduction is often causal and related to counterfactuals. It is therefore important that object identification and other machine-based

11Taken by Matthew Bisanz and presented as in Davis and Marcus (2015) to illustrate the many types of inferences that humans draw from stimuli.

12Faster R-CNN architecture (Ren et al., 2017) with RexNeXt-1010-32x8d backbone (Xie et al., 2017) and a Feature Pyramid Network (FPN), trained for three times the default number of iterations (3x schedule) and optimized for object detection on the COCO dataset (Lin et al., 2014).
inferences that are made from the data are based upon neural networks whose projections are robust to this form of out-of-distribution generalization.

Many facts about the world and logical relationships have been codified and introduced into AI models through the development of knowledge graphs, as illustrated by Figure 13. The diagram taken from the website of the ConceptNet open source database (Speer et al., 2017; Speer and Havasi, 2012), characterizes some key features about that particular database in its own lexicon. The directed graph consists of triplets of {subject, relationship, object} that represent abstract relationships such as “similar to” and “part of” between pairs of nouns. Much of the database is manually coded through crowdsourcing, an approach that has been found to produce less voluminous but more reliable data than those obtained through web crawling (Davis, 2017). ConceptNet and other knowledge graphs are frequently employed in the construction of vector-based embeddings for words, as in BERT or other transfer learning strategies, to enable the incorporation of these logical relationships into network learning approaches. Graph-based representations of knowledge are used in a variety of commercial applications including web search and question answering tasks by Google (Dong et al., 2014).

For queries whose answers already appear as entries in the database, usage of the knowledge graph is straightforward. For cases in which the answer is not identical to an existing entry, strategies for applying knowledge graphs are similar in general to those required for expanding the databases. As Ji et al. (2022) note, some graph-based approaches expand their knowledge bases through logical inference from existing relationships. However, the rate of exceptions to general logical rules, however, hinders the automation of that task. Consequently, network-based learning approaches are used on their own or supplemented with human verification. These networks take as a starting point embeddings produced from the knowledge graph and employ separate classification approaches to detect relevant nouns, pairs of nouns to link, and relationships among them (Ji et al., 2022; Wang et al., 2017). A thorough review of network-based knowledge graph approaches appears in Ji et al. (2022). Additional reviews discuss the construction and application of the graphs (Davis, 2017), the use of knowledge-enriched embeddings (Wang et al., 2017), and a quantitative comparison of approaches (Lin et al., 2018).

Competing approaches for representing semantic knowledge are evaluated annually in the SemEval competition. Tasks include multilingual detection of idioms and social cues such as a condescending or sarcastic tone and the parsing of taxonomies that are assumed or implied in context (Emerson et al., 2022). One example question from a prior year describes an event in which people sat down in a sauna and asks the artificial system whether they sat down on a bench or on the floor. The system is required to make this conclusion based upon outside knowledge of how a sauna is typically laid out. Highlights from the 2022 competition are described in Table 11. Selected results (winner, baseline, and in one case, human benchmark) are shown for Task 7 (the winner for most innovative task) and Task 11 (including the winner for the most innovative model) are shown. Results are shown for the top performer and the baseline for each task. The first of the two tasks is a fill-in-the-blank exercise similar to that performed in the training of BERT. The masked word is not obvious from the sentence, however, and must be inferred from the broader context of nearby sentences or from general knowledge, at times requiring understanding of subtle linguistic patterns such as metonymy. The human benchmark can complete the task with 79.4% accuracy. A BERT approach without supplemental knowledge performed at 45.7% accuracy. The winning approach by Shang et al. (2022) is an ensemble strategy that combines BERT-style network forecasting with the ERNIE knowledge graph-trained embeddings (Sun et al., 2021), which performs at 68.9%.

The second of the two tasks with results shown in Table 11 is to identify named entities from short snippets of text with little context—where the named entity might be a phrase such as “Inside Out” or “To Kill a Mockingbird” that also happens to be an artistic work. The learner must separately tag people, locations, corporations, other groups, products, and creative works and must do so in 11 different languages (the multi-language version of the test). The baseline BERT-based approach successfully completes the multilingual version of the task with 47.8% accuracy. The winning submission employs BERT-style embeddings together with a query-based knowledge retrieval approach, achieving 85.3% accuracy at the task.
7.2. Knowledge Extraction

In addition to incorporating domain knowledge and commonsense logic into deep learning systems, a growing body of research works to make neural networks’ outputs more easily interpretable. A human might justify the classification of an image by pointing to cues such as objects in Julia Child’s kitchen in Figure 12b or the mane and face of a lion. Researchers have observed, however, that a “neural-symbolic gap” exists, in the sense that neural networks’ outputs are generally difficult to distill into simple and explainable logical processes (Besold et al., 2017).

As Townsend et al. (2020) notes, some research aims to derive explanations of this form from black box systems like artificial neural networks. Craven and Shavlik (1995) describe the ex post translation of network-based decisions into tree-based logic. Guidotti et al. (2019); Park et al. (2018), and Pedreschi et al. (2019) develop strategies for considering counterfactuals, generalizing patterns into rules, and constructing text-based justifications for networks’ predictions. Work along these lines is highly in demand for application to real-world problems in medicine and finance. At the literature currently stands, however, no dominant tools exist to simply explain deep learning models’ projections, and it is a promising area for future research.

In the absence of systematic intuitive explanations that distill network-based knowledge, the deep learning community makes extensive use of techniques for identifying the relative importance of different predictors. Townsend et al. (2020) cover these methods in detail in their review. Some are discussed briefly below, with applications to some simple examples.

7.2.1. Shapley Values

One important technique through which machine learning researchers explain the importance of different model predictors is the use of Shapley values. The approach was initially developed by Shapley (1953) for use in the context of game theory. Its more recent use for model attribution in a machine learning context was promoted and popularized by Lundberg and Lee (2017). The approach is “model agnostic” and is compatible with settings like those in economics, finance, and medicine in which learning algorithms use diverse sets of predictors, any of which could conceivably be excluded from a given model specification. For a model with \( k \) predictors, the Shapley measure of influence for predictor \( j \) on a given test case is calculated by measuring its marginal contribution to the final classification—that is, the score if the predictor is included in the model minus the score if it is left out. This score is averaged over the \( 2^k - 1 \) ways in which subsets of the remaining \( k - 1 \) predictors are excluded from the model.

Shapley values have the advantage that they are highly versatile and can be used to understand any predictive model. Additionally, because the impact is averaged over multiple specifications, the statistic is robust to the many ways in which predictors interact and affect one another’s influence. While the original approach is computationally intensive, requiring \( 2^k \) trained versions of the model, the SHapley Additive exPlanations (SHAP) approach introduced by Lundberg and Lee (2017) employs approximations and other efficiency enhancements to make the calculation tractable. It does have the conceptual limitation, however, that its measure of a predictor’s influence is based in part upon what its contribution would be in some highly implausible model specifications.

Table 11

| Task                          | Example                                      | Model                          | Performance |
|-------------------------------|----------------------------------------------|--------------------------------|-------------|
| Multilingual Named Entities   | Patrick Gray, former director of the Federal Bureau of Investigation | DAMO-NLP - winning submission (Wang et al., 2020b) | 85.3%       |
| (MultiCoNER) (Maltasi et al., 2022) |                               | ROBERTa Baseline (Conneau et al., 2020) | 47.8%       |
| Plausible Clarifications (Roth et al., 2022) | Check ratings of different salons. Visit the salon’s website. Call ___ and ask questions. | Human benchmark (Roth et al., 2022) | 79.4%       |
|                               |                               | X-PuDu - winning submission (Shang et al., 2022) | 68.9%       |
|                               |                               | BERT Baseline (Devlin et al., 2019) | 45.7%       |

Performance on Selected Semantic Inference Tasks in SemEval-2022 Competition

Figure 14 presents Shapley values for one of the example natural language processing problems illustrated in Section 5. The sentence, shown along the vertical axis of Figure 14a is the first sentence of The Gettysburg Address (Lincoln, 1863), with the word “new” masked. A BERT model is used to predict the masked word. Figure 14b shows the model’s top five selections and their corresponding probabilities. SHAP values are computed for each of the words in the sentence, the punctuation, and the tokens [CLS], [MASK], and [SEP], which represent the beginning, position of masked word, and end of the sentence, all of which capture elements of the sentence itself and the context in which it appears (in this case as a standalone sentence). A custom gradient-based implementation is used here, with each word’s contribution aggregated across all of the hidden layers in the network.

As Figure 14b shows, the BERT selects “great” as the most likely word to appear in place of the [MASK] token. The correct value of “new” is the second choice with roughly equal probability to “united.” As Figure 14a shows, the SHAP values provide some insight into the transformer’s decision-making process. The components of the sentence that are most supportive of the top choice of “great” are the position of the masked word and “dedicated,” and the component that is least supportive of “great” is the beginning, which is generally interpreted as the context in which the sentence appears. The word “new” is relatively uninformative, being redundant with “brought forth.” Relative to the first choice, “new” is driven less by
Generalization in neural networks

7.2.2. Attribution Heatmaps

While SHAP values are sometimes applied to images, some heatmap-based approaches are popular and informative for deep convolution networks in vision-related tasks. Figure 15 illustrates two methods for the attribution of network-based features to model outputs. Both are applied to better understand the classifications of the artificially generated lion photos presented in Section 3. Results are shown for the MobileNet v3 Large network, as that was one for which “lion” was predicted with high confidence when the animal was in its natural habitat and “chow chow” was predicted, also with high confidence, when the animal was observed in the city. The images help to illustrate what factors led the network to arrive at those conclusions. All four images show pixel-specific heatmaps illustrating what portions of the images are important for the final projections. In all cases, the heatmaps are superimposed on the original images for readability.

The first approach, Layer-wise Relevance Propagation (LRP), introduced by Bach et al. (2015), considers attribution in a holistic way. For each pixel’s contribution to the confidence score for the chosen class is summed over all the layers in the neural network. The LRP thus indicates how much each of the pixels in the original image matters in the model’s final classification.

The second approach, gradient-weighted Class Activation Mapping (gradient CAM), was developed by Selvaraju et al. (2017). It constructs a heatmap at the pixel level based upon the last convolutional layer in the model. The intensity illustrated in the map is the gradient of that pixel on the confidence score for the chosen class. Thus the gradient CAM shows, after the many network steps, what elements in the final processed image are most important for its classification.

As the graphs illustrate, the model is more concentrated on the object of interest in the final convolution layer than overall. Additionally, the many distracting elements in the city photograph inhibit the model’s ability to focus on the object of interest. The model’s attention is diluted in the city photo as compared to the natural habitat photo.

For the lion in its natural habitat in Figure 15a in the top left, the MobileNet v3 Large network dedicates most of its attention to the upper and central portions of the image, and it is those areas that contribute the most to the model’s final classification. The model makes relatively little use of the information from the grassy areas. These low scores are consistent with the background being less important than the foreground in identifying the object in the photograph. As Figure 15c below it shows, the focus on the foreground is more pronounced in the final convolution layer. While the earlier layers of analysis use a larger part of the image to identify edges and shapes, by the time the analysis reaches the final convolution layer, the network’s attention is concentrated on the parts of the photograph that are most relevant for discerning the class. In the case of the lion in its natural habitat, that area is the animal’s face.

On the right-hand side, Figures 15b and 15d illustrate a similar pattern, but with some important differences. The LRP, which captures influence throughout the layers of the network, shows a wide dispersion of influence across the photograph. While the area containing the lion itself matters the most, the street, van, and buildings all have notable contributions to the final classification. As with the natural habitat photo, moving from LRP to gradient CAM increases the amount of attention that is focused on the subject in the foreground. The level of focus is considerably less concentrated in the city photo than in the natural habitat photo, however. The many city-related elements in that picture appear to be key contributors to its final selection of chow chow as the class for that image.

Figure 14: Results from BERT Analysis of First Sentence of Gettysburg Address with “new” Masked

other words in the sentence and is more sensitive to the positional terms and punctuation.
In order to more clearly illustrate the tradeoffs that the classifier makes in its considerations, Figure 16 shows gradient CAM representations not only for the chosen class but for the model’s second and third choices for each image. These heatmaps help to illustrate what portions of the photograph made the chosen class more likely as compared to which portions mattered for the key alternatives. For both images, the most promising choices besides lion are dog breeds.

For both photographs, the components of the image that contribute most strongly to a classification of lion—shown in the first CAM heatmap in Figure 16a and the second in Figure 16b—are concentrated around the animal’s face. As the other heatmaps show, for both images, the legs and surrounding area are more important contributors to the classification of the image as a breed of dog.

8. Abstraction in Nature

As researchers endeavor to improve the ability of artificial networks to generalize and abstract, the structures of animal brains provide a valuable template. One distinguishing feature of biological neural networks is modularity. Neuroscientists have found that animal brains are segmented into well-defined components or modules that are specialized for the handling of specific tasks. In Drosophila, for instance, standardized innate responses are processed in the lateral...
horn, while learned responses and prioritization are handled by a separate component known as the mushroom body (Aso and Rubin, 2016; Aso et al., 2014; Devineni and Scaplen, 2022; Rohlfs, 2023a).

This modular structure is illustrated in a stylized form in Figure 17a, which is a schematic interpretation the structures in animal brains through which data are routed to the appropriate type of response. Each sensory input—described in the diagram as “new data” is routed by a dispatcher into one of three channels: innate response, learned response, and more data. The network of evolved pathways, with separate treatment for distinct forms of inputs, can be thought of as the dispatcher. Stimuli that have hard-wired reactions are often handled in a separate component of the brain. Modularity of this form has been observed in a variety of animals, with different parts of the brain activated during abstract tasks than concrete ones in humans (Gilead et al., 2014; Wurm and Lingnau, 2015; Vaidya et al., 2021), and lesions to the prefrontal cortex have been found to inhibit abstract thinking in primates (Mansouri et al., 2020). This processing is often viewed as hierarchical, with successively complex components of the brain handling successively sophisticated problems (Kaiser et al., 2010; S. J. Kiebel and Friston, 2009; Meunier et al., 2009, 2010).

The organization of the brain into specialized components helps it to maintain consistent performance across contexts. It helps to diversify risks by containing the overall system’s vulnerability to any one component’s errors. Because each module is used in a variety of applications and is informed by observations from those contexts, the trained structure is relatively insensitive to aberrations in any one domain of knowledge (Sinz et al., 2019). Stability is also maintained in the brain through system-wide limits on the potential influence of new information. Sadeh and Clopath (2021) describe the brain’s process of inhibitory stabilization, whereby a given component’s signal expression might be subdued by other parts of the brain that have a broader perspective on the priorities of the overall structure. In addition to this short-term regulation of signal strength, brains perform a process of controlled forgetting of longer-term information that is influenced by the rates at which new neurons are created (Akers et al., 2014). This gradual removal of outdated information enables the system to adjust to general trends that impact relationships and categories in the data without the thrashing from one decision rule to another that could emerge if updating were instantaneous (Richards and Frankland, 2017).

The routing process illustrated in Figure 17a takes into account the priorities of the organism according to multiple time scales. If recent events call for immediate action—e.g., flight—then chemical signals produce a temporary change in overall priorities at the system level (Devineni and Scaplen, 2022). Savin and Triesch (2014) find that, when a primate moves from one task to another, the change is evident through a shift in activity among large numbers of neurons in the prefrontal cortex—with urgent tasks causing larger numbers of cortical neurons to be repurposed. Also among primates, Froudist-Walsh et al. (2020) note that system-wide dopamine levels in the brain impact the performance of working memory—in some cases promoting recurrence.

---

**Figure 16:** Gradient CAMs for Alternate MobileNet v3 Large Classifications
Generalization in neural networks

| Response Type       | Storage Location                                      | Means of Coding                                |
|---------------------|-------------------------------------------------------|------------------------------------------------|
| Innate              | Neuroanatomic structure and connectivity patterns     | Persistence of genetic code through evolution   |
| Learned             | Dedicated memory components                           | Dopaminergic conditioning                       |

(a) Module-based Routing

(b) Innate and Learned Responses

Figure 17: Responses to Stimuli in Animal Brains

among the affected sets of neurons or blocking inhibitory signals from other cells. From a long-term perspective, system-wide priorities are embedded into the structure of the network, with receptors to important stimuli exercising more influence in the network. Connections that appear to reflect and to impact subjective valuations of past and prospective actions are distributed throughout the brain (Lee et al., 2012). Ji et al. (2020) find evidence of circuit-wide activity patterns such as mating, foraging, exploration, exploitation that influence behavioral state for multiple components of the brain in roundworms.

The first two types of responses in Figure 17a—innate and learned—are fundamentally different in the ways in which they develop and operate. These differences are summarized in Figure 17b, which describes how innate and learned responses differ in the ways in which the information is coded and stored. The third response, “more data,” reverts back to the top of the diagram, so that the responses described in Figure 17b are the absorbing states of the flow chart.

Like the overall modular design and routing process, innate responses are driven by structural characteristics—the overall layout of the neural network as well as key hard-wired responses—that are stable over the life of the organism. These rules are thus stored in DNA and originate and change slowly through the process of genetic evolution. Barabási and Barabási (2020) find that large patterns of synaptic connections—patterns that they call bicliques—are often shared across different animals of the same species. The authors propose that these similarities are innate, and they introduce a model to identify which portions of the connectome are coded by common genes. As Zador (2019) notes, the amount of information stored as innate responses is minuscule compared to the overall capacity of the brain: the human genome can encode at most about a gigabyte of information relative to the 500 terabytes of information contained in the human brain connectome.

“Learned,” the second response type in Figure 17, relates to memory and plasticity, which serve an important role in the modular approach to thinking. In the case of Drosophila, Vogt et al. (2014, 2016) find that the structure through which neural pathways group sensory inputs into categories is largely fixed and that a key building block to the formation of memories is the learning of positive and negative associations with those categories. Depending on the flies’ experiences, a given odor might attract some flies and repel others. The learned associations that are stored in memory are different, but the mental architecture mapping the stimulus to a response is similar. Knoblauch et al. (2010) characterize the first type of learning—the formation of positive or negative associations—as synaptic plasticity. A second and less common type of learning, which Knoblauch et al. (2010) refer to as structural plasticity, involves the modification of the categories with which associations are formed. In one example among fruit flies, Ueno et al. (2017) find a form of this plasticity that is a relatively sophisticated process that is only activated if both visual and olfactory sensory signals have been received—so that sensory input from one source serves to verify input from another source.

The grouping of stimuli in memory is a vital element of abstract thought. Neuroscientists find that the amount of abstraction and the level of complexity of these mappings are determined by the importance of such categories to the broader goals of the organism (Cortese et al., 2021; Stegmann et al., 2020; Timme et al., 2016). Rewards and penalties that are regarded as salient are connected to dopamine signals that drive animals to learn about causal relationships and the world around them. Animals acquire feedback from experience regarding the accuracy of the learned connections between stimuli and rewards or penalties—and the timing and intensity of neurons’ dopamine responses update and adjust accordingly (Kahnt and Tobler, 2016; Schultz et al., 1997; Schultz, 1998). A variety of intricate associations are are coded through this...
producing previously encountered examples from the same category, have been found to influence how new stimuli are perceived. (Feldman et al., 2009), the categories developed in memory (DiCarlo, 2017; Min et al., 2020) and listening in humans experience of sensation. In experiments of vision in primates to produce archetypal examples that then interact with the general rules. memories serves an important role in this process of Robertson (2018) argues that the instability of specific dependence (Dayan and Abbott, 2001; Schultz, 2007). Time scales and accommodating uncertainty and context-dopamine-guided learning process, operating on different time scales and accommodating uncertainty and context-dependence (Dayan and Abbott, 2001; Schultz, 2007). Robertson (2018) argues that the instability of specific memories serves an important role in this process of abstraction, enabling the translation of specific events into general rules.

This mapping of sensory inputs helps animal brains to produce archetypal examples that then interact with the experience of sensation. In experiments of vision in primates (DiCarlo, 2017; Min et al., 2020) and listening in humans (Feldman et al., 2009), the categories developed in memory have been found to influence how new stimuli are perceived. Gaps in sensory information are filled in based upon previously encountered examples from the same category, producing a perceptual magnet effect, which allows the brain to exploit existing trained skills and mental architecture that are built around these categories. These archetypes also assist in the perception of objects that are partially obscured, are placed in unusual contexts, or have distracting elements. DiCarlo (2017) notes that, for the first 100 milliseconds of object detection, artificial classifiers’ facilities rival those of humans and primates. When faced with more difficult cases in which the solution takes additional time, however, the biological brains’ advantage becomes more pronounced.

This perceptual magnet effect is illustrated in Figure 18. In the image, the viewer perceives the left wing of a butterfly, but the body and right wing are obscured by the tree trunk and consequently unseen. In perceiving the butterfly, the viewer’s mind matches the visual data to a category in memory and fills in the missing information based upon an image stored in memory of what a typical case looks like. This form of information processing helps humans to easily solve CAPTCHA-type problems such as those discussed in Section 4 and illustrated in Figure 6.

While the brain may initially fill in gaps based upon archetypes, both synaptic and structural plasticity interact with the third response from Figure 17a, the collection of new data. The pursuit of new data for learning is online and curiosity-driven. Biological systems select cases to learn based upon what is new, deviates from expectation, or fills specific gaps in understanding (Sinz et al., 2019). This targeted training helps animals to limit cognitive effort to cases that provide sufficiently high benefit. A certain training case might offer insights into one chunk of a tasks but not another. Separating the information into chunks helps to identify what parts of a training example are new and to confine relearning and updates to those sub-problems for which the observation is relevant.

Research by Savin and Triesch (2014) suggests a practical set of principles for the organization of conceptual “chunks” in primates. Events are grouped into similar categories in working memory if they are connected to the same external stimulus, resulting motor response, or some combination of the two. Cohen et al. (2021) similarly argue that practice and repetition are performed in animals in order to achieve a consistent motor response. Additionally, research by Logiaco et al. (2019) suggests that animal brains may learn complex motor patterns by breaking them down into manageable “motor motifs.”

Through this process, the construction of concepts and memories is driven by the same forces that motivate the organism more broadly. Nevertheless, the parts of the brain that are responsible for abstraction behave in many ways as distinct modules that operate on their own time scale. While the learning process is initially motivated by the pursuit of dopaminergic rewards, incentives and learning are distinct in the brain, and knowledge that is acquired in order to satisfy a previous need persists when priorities change (Berridge, 2012; Berridge and Robinson, 2016; Zhang et al., 2009). Hence, “motivational salience,” which is malleable and based upon recent situational data, drives an animal’s behavior. Its actions are also informed, however, by facts and

| Response Type                | Feature                  | Description                                                                 | Artificial Analogues                                                                                       |
|------------------------------|--------------------------|------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| Modularity and Innate Responses | Specialized Components   | Different anatomical components specialized to handle specific tasks          | Transfer of feature transformations and embeddings (Sections 4 to 7), Evolutionary deep learning (Baymurzina et al., 2022; Wang et al., 2020a; Zhan et al., 2022) |
| System-wide Prioritization  | Domain generalization and concept drift (Section 4), Chemical signals regulate entire system behavior based upon context | Inhibitory signals (Cao et al., 2018; Seung, 2018a,b), Multi-objective optimization (Kim et al., 2022; Qu et al., 2021) |
| Learning and Abstraction     | Archetypes               | Repeated responses saved to conserve computational resources                 | Feature embeddings (Sections 4 to 7), Knowledge Distillation (Gou et al., 2021; Ku et al., 2020), Neural architecture search (Baymurzina et al., 2022; Dokeroglu et al., 2022; Jaafra et al., 2019) |
| Synaptic Plasticty           | Prior experiences and responses associated with archetypes through conditioning | Feature embeddings (Sections 4 to 7), Meta-learning (Section 5)                  |
| Structural Plasticity        | Representations adapt over time based upon relevance                        | Meta-learning (Section 5), Concept drift and domain generalization (Section 4) |
| Semantic Knowledge           | Use of facts for deductive reasoning                                        | Knowledge graphs (Section 7)                                                                 |
| Data Collection              | Selective Attention       | Salient events receive more granular representations and more data collected  | Replay and practice (Andrychowicz et al., 2017), Data Augmentation (Shorten and Khosigofaar, 2019; Tian and Zhang, 2022), Adversarial training (Chen et al., 2020; Frolov et al., 2021; Lust and Condurache, 2021; Qiu et al., 2022), Self-paced or curriculum learning (Soviany et al., 2022) |

Table 12
Artificial Analogues to Biological Modularity
understanding about the world that have been accumulated over its lifetime.

Once such concepts are established, an essential component of the ability to generalize is the identification of relationships among these categories. As a motivating example, Amidzic et al. (2001) find evidence of “chunking” in the neural activity of chess grandmasters. The authors observe that, when playing chess against a computer, grandmasters make less use than novices do of portions of the brain associated with analysis and recent memories, and they make greater use of parts of the brain associated with long-term memory. Thus the key difference in mental activity that the authors observe between experts and novices is not in the experts’ use of faster or more powerful processors. The experts’ brains employ principles that have been distilled and stored from previously encountered cases and tools for recognizing which principles apply when faced with a new situation.

Table 12 briefly summarizes some of the ways in which modularity can be seen in biological systems and analogous strategies used by artificial neural networks. Many of these approaches are discussed earlier in this paper. For artificial approaches that employ elements of modularity but do not relate directly to generalization and the scope of this paper, a few citations are included as pointers for the interested reader.

The first category of response types described in Table 12, “Modularity and Innate Responses,” is broken into two features and relate to the role of anatomy in the modular behavior of neural networks. As noted in the discussions of Figure 17 above, the routing of stimuli into innate and conditioned responses occurs through the use of components of the brain that are specialized for those tasks—whereby the innate responses emerge slowly through evolution while conditioned responses develop over the life of the organism. This distinction, in which different types of responses follow different training processes and timelines, resembles the use of feature transformations in transfer learning models as well as in with pre-trained embeddings such as BERT and knowledge graphs that are developed in a general setting and then deployed for use on specific applications. Experimentation and variation based upon the principle of genetic recombination can also be seen in evolution-inspired processes for optimization of complex systems.

The second feature listed in the table, “System-wide Prioritization,” relates to the manner in which the full cognitive apparatus is equipped to respond to large scale events that impact the tradeoffs faced by individual components. Such behavior in artificial neural networks is discussed in Section 4 above on domain generalization. Discussions in the literature on multi-objective optimization—and preferred methods for balancing different goals—offer another perspective on this prioritization problem. Additionally, some approaches simulate the strategy used in animal brains in which neurons send inhibitory and excitatory signals to regulate and balance the impacts of stimuli of different types.

The next category of response types, “Learning and Abstraction,” involves modularity in the grouping of combinations of stimuli into ideas. The use of archetypes in the brain is mimicked in artificial systems through the forms of generalization described in Sections 4 to 7, including the use of feature embeddings. Related work in artificial neural networks that does not explicitly generalize but performs some manner of abstraction involves knowledge distillation—through which smaller networks are trained to exhibit the same behavior as larger and more complex ones. The literature on neural architecture search involves a somewhat different take on abstraction, in which decisions about the structure of the network are assisted through explicit optimization, so that the classifier takes on some of the responsibility that is traditionally handled by the researcher.\textsuperscript{13}

\textsuperscript{13}Results from some search-based network classification approaches appear in the discussion of overfitting in Table 2.
“Synaptic plasticity,” the learning of positive or negative associations with existing archetypes, is achieved in artificial systems both through feature embeddings and meta-learning, both of which take network structures developed in one setting and apply them to other problems. Strategies related to “structural plasticity,” the development and refinement of network-based architecture for learning such associations, involve meta-learning as well as transfer- and drift-based domain generalization approaches. Another key form of abstraction, “Semantic Knowledge,” relates to knowledge graphs and scope generalization, as discussed in Section 7.

In relation to the third type of biological response from Figure 17b, the selective collection of additional data, a variety of deep learning approaches augment existing datasets through regularization or adversarial approaches, and a recent area of growth known as self-paced or curriculum learning presents training examples to a network in increasing order of difficulty.

9. Discussion and Conclusion

This paper reviews concepts, modeling approaches, and recent findings related to the generalization of insights from neural networks. A taxonomy of forms of generalization is introduced, with a spectrum of increasing levels of abstraction, including generalization across (1) Samples, (2) Distributions, (3) Domains, (4) Tasks, (5) Modalities, and (6) Scopes. For (1) sample generalization from training to test data, theoretical conditions for generalizability, regularization strategies to improve robustness, and empirical results from computer vision are discussed. Suggestive results indicate that, in the case of ImageNet, popular models exhibit substantial amounts of overfitting, with training accuracy 20 percentage points or more higher than accuracy on a separately provided test sample of images.

Statistical views of (2) distribution generalization are reviewed, including the importance of datasets that make it possible to identify causal relationships and counterfactuals. An empirical example from computer vision is introduced of a lion pictured in an urban environment. Some established neural networks trained on ImageNet incorrectly classify the animal as a dog, highlighting the extent to which network-based training relies in some cases on confounding variables, such as the surrounding environment in a photograph. These variables, which may be deep or unobserved, are correlated with target classes in training, and models in some cases learn unstable relationships driven by those predictors that fail to generalize across populations and to counterfactual cases.

Influential frameworks on (3) domain generalization are summarized, including concept drift as well as transfer learning to adapt to a shift in domain. The Photo-Art-Cartoon-Sketch (PACS) dataset is presented as a motivating example. Recent progress is discussed as are a variety of different domain generalization datasets that are used as benchmarks in the literature. The importance of the transfer of learned structures and feature embeddings is highlighted as a key strategy for generalization across domains and higher levels of abstraction.

Recent (4) task generalization breakthroughs are discussed in the areas of few-shot meta-learning—in which the skill of object identification is expanded by introducing new object classes to identify, the multipurpose BERT approach to NLP, and the growing use of transformer-based foundation models that can be applied to many tasks with minimal retraining.

Generalization across (5) modalities remains a relatively unexplored area, but recent innovations include the use and rising popularity of text-based embeddings in image classification and the transfer of a biologically-inspired network across olfactory, visual, and auditory modalities.

Exciting developments in (6) scope generalization involve supplementing deep NLP approaches with embeddings based upon graphical representations of symbolic and semantic knowledge. State-of-the-art performance from the SemEval competition is discussed. Additionally, attribution tools for extracting knowledge from deep learning models are illustrated using examples from NLP and computer vision.

The recent rapid advances that have been seen in different forms of generalization—particularly across domains and tasks—help to underscore the considerable power that deep, network-based predictive systems have demonstrated. Domain generalization, few-shot learning, and the emergence of versatile transformer based foundation models in NLP are possible in part because artificial neural networks learn quickly, generalize well both out-of-sample and out-of-distribution, and perform effectively at a wide range of discriminative and generative tasks. The speed with which these systems have been emerging and improving highlights the many directions in which further developments can be expected in the coming years.

Even with such fast and consistent evolution across so many dimensions, the suggestive results introduced in this article on sample and distribution generalization draw attention to some potential directions for future research in which there may be some room for improvement. When applying well-known neural networks to classify ImageNet images, a consistent gap is apparent between the training data and a separately provided test sample. Additionally, half of those models failed to correctly recognize a lion when it was moved from its natural habitat into an urban environment. Both of these suggestive results pertain to just one dataset and application. Given, however, the considerable degree to which diverse applications rely upon backbone networks such as these, the possibility that the stability and robustness of their forecasts could be improved upon is a topic worthy of further investigation.

In addition to these patterns and findings that concern specific ways in which models models are generalized, this article provides insights about abstraction in neural networks more generally and how biological and artificial approaches to these tasks are related. The later part of the survey explores neuroscience literature on how biological...
brains perform generalization and abstraction tasks, focusing on their modular structure as well as the dopamine-driven organization of learned ideas into conceptual “chunks.”

The widespread adoption of transformer-based foundation models is one important way in which AI systems are moving in the direction of this modular approach to intelligence. Tasks like recognizing objects in images or understanding text are effectively standalone. These basic problems are effectively solved, and researchers working on specialized offshoots of that work no longer have to reinvent the wheel. Each time that marginal improvements are made to those solutions, they are swiftly deployed, and that learning is leveraged at a global scale.

As foundational models are increasingly used as modules of this form, some elements could help them to move closer to the parsimonious and efficient architecture of brains. It is desirable, for instance, to make the modules themselves highly stable and robust to generalization across many populations and domains. Training the systems with diverse sets of unusual and counterfactual cases may help in those dimensions.

Additionally, brains process information hierarchically, with simple and common cases handled immediately and complex and rare cases taking additional effort. Artificial systems could in principle adopt a similar structure, breaking up the larger models into submodules that use routing rules to handle new cases. Research in few-shot learning shows that very small submodules can produce accurate projections. Designs of this form could help to improve total system stability, by preventing aberrations from exercising outsized influence on the overall system. At the same time, having submodules that work with small numbers of rare cases would enable plasticity-like updating at a decentralized level to events such as concept drift. Such updating could use incoming data to fill in gaps in knowledge, like in biological brains’ curiosity-driven learning.

The movements in AI toward multimodal analysis and generalization across modalities also point toward a convergence between artificial and biological systems. Evidence from the neuroscience literature suggests that some important forms of learning and memory are multimodal, with internal representations applying to abstract entities rather than looks, sounds, or other specific sensory indicators. Publicly available foundation models are becoming the memory modules of the world’s AI systems. Systems that make greater use of multimodal inputs and recognize connections among them make help to move those memory modules closer to the concept-based thinking of animals and humans.

An additional feature of biological systems that may be used fruitfully in AI systems is perceptual magnet-style inference from limited information. Researchers have found that synthetic data such as machine-generated captions on images (Vasu et al., 2024) and wins and losses from self-play of games (Silver et al., 2017) can substantially improve a model’s performance in real-world situations. Work in AI has approached this topic by feeding simulated data directly into learning-based systems as training data. In biological brains, similar tasks are performed automatically through the use of archetypes stored in the brain. One promising area of research could be to find ways in which artificial neural networks could make such connections immediately without those extra steps.

In the reverse direction, some of the questions that arise in deep learning suggest possible avenues for neuroscience research. While brains are known to exercise selective, curiosity-driven attention, understanding the process through which new data are considered or ignored and the importance of dopamine in driving this process may help to inform biologists’ understandings of neuroplasticity. Additionally, research into how brains divide into modules—which tasks are handled by which components and how each of them update according to new sensory input—could shed light on how learning occurs at different levels and how brains make the transition from the specific to the abstract. Additionally, some of the tools that deep learning researchers use for knowledge extraction and attribution have the potential to provide insights to the increasingly complex computational models that neuroscientists are developing to represent brain activity.

As the neural network community continues to explore and progress, we can expect artificial systems to continue to become more stable and robust and to achieve broader and more consistent levels of abstraction. The experiences seen in applying artificial neural networks at a spectrum of different levels of generality—and the insights that the field of neuroscience offers into analogous biological processes—both provide valuable blueprints of potential paths forward.

References

Addepalli, S., Asokan, A.R., Sharma, L., Babu, R.V., 2024. Leveraging vision-language models for improving domain generalization in image classification. Computer Vision and Pattern Recognition, 23922–23932.

Akers, K.G., Martinz-Canabal, A., Restivo, L., Yiu, A.P., De Cristofaro, A., Hsiang, H., Wheeler, A.L., Gusjkolen, A., Niibori, Y., Shoji, H., Ohira, K., Richards, B.A., Miyakawa, T., Joseelyn, S.A., Frankland, P.W., 2014. Hippocampal neurogenesis regulates forgetting during adulthood and infancy. Science 344, 598–602.

Akhbar, Z., Dasgupta, D., 2019. A brief survey of adversarial machine learning and defense strategies. University of Memphis Technical Report CS-19-002.

de Albuquerque Teixeira, R., Braga, A.P., Takahashi, R.H.C., Saldanha, R.R., 2000. Improving generalization of mlps with multi-objective optimization. Neurocomputing 35, 189–94.

Amidzic, O., Riehle, H.J., Fehr, T., Wienbruch, C., Elbert, T., 2001. Pattern of focal γ-bursts in chess players. Nature 412, 603.

Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M.W., Pfau, D., Schaul, T., Shillingford, B., de Freitas, N., 2016. Learning to learn by gradient descent by gradient descent. Advances in Neural Information Processing Systems (NIPS) 30, 3988–96.

Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Abbeel, P., Zaremba, W., 2017. Hindsight experience replay. Advances in Neural Information Processing Systems (NIPS) 31, 5055–65.

Angrist, J.D., Pischke, J., 2009. Mostly Harmless Econometrics. Princeton University Press, Princeton, NJ.

Arpit, D., Wang, H., Zhou, Y., Xiong, C., 2022. Ensemble of averages: improving model selection and boosting performance in domain
Li, P., Wang, J., Zeng, Z., 2022a. An overview of the stability analysis of recurrent neural networks with multiple equilibria. IEEE Transactions on Neural Networks and Learning Systems forthcoming.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swim transformer: hierarchical vision transformer using shifted windows. International Conference on Computer Vision.

Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S., 2022b. A convnet for the 2020s. arXiv preprint https://arxiv.org/abs/2201.03545.

Logiaco, L., Abbott, L.F., Escola, S., 2019. A model of flexible motor sequencing through thalamic control of cortical dynamics. bioRxiv preprint doi: http://dx.doi.org/10.1101/2019.12.17.889153.

Louizos, C., Shahil, U., Mooij, J., Sontag, D., Zemel, R., Welling, M., 2017. Causal effect inference with deep latent-variable models. Neural Information Processing Systems 30.

Lu, J., Liu, A., Dong, F., Gama, J., Zhang, G., 2019a. Learning under concept drift: a review. IEEE Transactions on Knowledge and Data Engineering 31, 2346–63.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, H., Zhang, G., 2019b. Learning under concept drift: a review. IEEE Transactions on Knowledge and Data Engineering 31, 2346–63.

Lundberg, S.M., Lee, S., 2017. A unified approach to interpreting model predictions. Neural Information Processing Systems 30, 4785–74.

Lust, J., Condurache, A.P., 2021. A survey on assessing the generalization envelope of deep neural networks: predictive uncertainty, out-of-distribution and adversarial samples. arXiv preprint https://arxiv.org/pdf/2008.09381.pdf.

Malmasi, S., Fang, A., Fetabu, B., Kar, S., Rokhlenko, O., 2022. SemEval-2022 task 11: multilingual complex named entity recognition (multicner). International Workshop on Semantic Evaluation (SemEval) 16, 1412–37.

Mangla, P., Kumari, N., Sinha, A., Singh, M., Krishnamurthy, B., Balasubramanian, V.N., 2020. Charting the right manifold: manifold mixup for few-shot learning. IEEE Winter Conference on Applications of Computer Vision , 2218–27.

Mansouri, F.A., Freedman, D.J., Buckley, M.J., 2020. Emergence of abstract rules in the primate brain. Nature Reviews: Neuroscience 21, 595–610.

Mao, X., Chen, Y., Jia, X., Zhang, R., Xue, H., Li, Z., 2022. Context-aware robust fine-tuning. arXiv preprint https://arxiv.org/abs/2211.16175.

Manzoni, G.C., Petrini, A., Malchiodi, D., Frasca, M., 2023. Deep neural network compression: a comparative survey and choice recommendations. Neurocomputing 520, 152–70.

Masci, J., Meier, U., Ciresan, D., Schmidhuber, J., 2011. Stacked convolutional auto-encoders for hierarchical feature extraction, in: Honkela, T., Duch, W., Girolami, M., Kaski, S. (Eds.), Artificial Neural Networks and Machine Learning – ICANN 2011. ICANN 2011. Lecture Notes in Computer Science, vol 6791. Springer, Berlin, pp. 52–9.

Meunier, D., Lambiotte, R., Bullmore, E.T., 2010. Modular and hierarchically modular organization of brain networks. Frontiers in Neuroscience 4, 200.

Meunier, D., Lambiotte, R., Fornito, A., Ersche, K.D., Bullmore, E.T., 2009. Hierarchical modularity in human brain functional networks. Frontiers in Neuroinformatics 3, 37.

Meunier, D., Lambiotte, R., A., Freedman, D.J., Wang, X., 2020. A neural circuit mechanism of categorical perception: top-down signaling in the primate cortex. bioRxiv preprint doi: doi:10.1101/2020.06.15.151506.

Mitchell, T.M., 1997. Machine Learning. McGraw-Hill, New York, NY, USA.

Mohri, M., Rostamizadeh, A., Talwalkar, A., 2018. Foundations of machine learning. The MIT Press, Cambridge, MA.

Moradi, R., Berangi, R., Minaei, B., 2020. A survey of regularization strategies for deep models. Artificial Intelligence Review 53, 3947–86.

Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., Sutskever, I., 2021. Deep double descent: where bigger models and more data hurt. International Conference on Learning Representations.

Neelakantan, A., Vilnis, L., Le, Q., Sutskever, I., Kaiser, L., Kurach, K., Martens, J., 2015. Adding gradient noise improves learning for very deep networks. arXiv preprint https://arxiv.org/abs/1511.06867.

Neyshabur, B., Bhojanapalli, S., McAllester, D., Srebro, N., 2017. Exploring generalization in deep learning. Neural Information Processing Systems (NIPS) 31, 5949–58.

Niu, B., Wen, E., Ren, W., Zhang, X., Yang, L., Wang, S., Zhang, K., Cao, X., Shen, H., 2020. Single image super-resolution via a holistic attention network. European Conference on Computer Vision (ECCV), 191–207.

OpenAI, 2024a. A candid photo of a lion in its natural habitat, resembling a tourist snapshot. the lion is walking through tall grass, with some bushes and sparse trees. Digital image. Created using DALL-E.

OpenAI, 2024b. A candid, photorealistic picture of an empty street in midtown manhattan, new york, with a scruffy, imperfect lion standing right next to a white van. Digital image. Created using DALL-E.

Pandit, T., Kudithipudi, D., 2020. Relational neurogenesis for lifelong learning agents. Neuron-inspired Computational Elements Workshop (NICE).

Papers With Code, 2024. Domain generalization on pacs. https://paperswithcode.com/sota/domain-generalization-on-pacs-2.

Park, D.H., Hendricks, L.A., Akata, Z., Rohrbach, A., Schiele, B., Darrell, T., Rohrbach, M., 2018. Multimodal explanations: justifying decisions and pointing to the evidence. IEEE Conference on Computational Vision and Pattern Recognition , 8779–88.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: an imperative style, high-performance deep learning library. Neural Information Processing Systems 32, 8024–35.

Pearl, J., 2009. Causality. Cambridge University Press, Cambridge, United Kingdom.

Pedreschi, D., Giannotti, F., Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., 2019. Meaningful explanations of black box ai decision systems. AAAI/IAAI/EAAI Conference on Artificial Intelligence 33, 9780–4.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B., 2019. Moment matching for multi-source domain adaptation. International Conference on Computer Vision , 1406–15.

Peng, X., Usman, B., Kaushik, N., Hoffman, J., Wang, D., Saenko, K., 2017. VisDA: the visual domain adaptation challenge. arXiv preprint https://arxiv.org/pdf/1710.06924v2.pdf.

Peters, J., Janzing, D., Schölkopf, B., 2017. Elements of causal inference: foundations and learning algorithms. The MIT Press, Cambridge, MA.

Pirraccchio, R., Hubbard, A., Sprung, C.L., Chevret, S., Annane, D., 2020. Assessment of machine learning to estimate the individual treatment effect of corticosteroids in septic shock. JAMA Network Open 3, e2029050.

Pratama, M., Ashfahani, A., Hady, M.A., 2019. Weakly supervised deep learning approach in streaming environments. IEEE International Conference on Big Data, 1195–202.

Prechelt, L., 2012. Early stopping—but when?, in: Montavon, G., Orr, G.B., Muller, K. (Eds.), Artificial Neural Networks and Learning Systems forthcoming.

Papers With Code, 2024. Domain generalization on pacs. https://paperswithcode.com/sota/domain-generalization-on-pacs-2.

Press, O., Wolf, L., 2017. Using the output embedding to improve language models. European Chapter of the Association for Computational Linguistics 2, 157–63.

Qi, G., Aggarwal, C., Huang, T., 2011. Towards semantic knowledge propagation from text corpus to web images. International Conference on World Wide Web 20, 297–306.

Qian, Z., Huang, K., Qang, Q., Zhang, X., 2022. A survey of robust matching for multi-source domain adaptation. International Conference on Computer Vision 2021. Swim transformer: hierarchical vision transformer using shifted windows. International Conference on Computer Vision .
Generalization in neural networks

Quiroga, R.Q., Reddy, L., Kreiman, G., Koch, C., Fried, I., 2005. Invariant visual representation by single neurons in the human brain. Nature 435, 1102–7.

Raford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I., 2021. Learning transferable visual models from natural language supervision. arXiv preprint https://arxiv.org/abs/2103.09020.

Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P., 2020. Designing network design spaces. Computer Vision and Pattern Recognition, 10428–10436.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Raford, A., Chen, M., Sutskever, I., 2021. Zero-shot text-to-image generation. arXiv preprint https://arxiv.org/abs/2102.12092.

Recht, B., Roelofs, R., Schmidt, L., Shankar, V., 2019. Do ImageNet classifiers generalize to ImageNet? International Conference on Machine Learning 97.

Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster r-cnn: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 39, 1137–49.

Richards, B.A., Frankland, P.W., 2017. The persistence and transience of memory. Neuron 94, 1071–84.

Robertson, E.M., 2018. Memory instability as a gateway to generalization. PLOS Biology 16(3), e2004633.

Rocha, H.P., Costa, M.A., Braga, A.P., 2020. Neural networks multiobjective learning with spherical representation of weights. IEEE Transactions on Neural Networks and Learning Systems 31, 4761–75.

Rogers, T., 2014. Captcha font. http://cargocollective.com/

tomrogers/CAPTCHA.

Rohlfis, C., 2023a. A descriptive analysis of olfactory sensation and memory in Drosophila and its relation to artificial neural networks. Neurocomputing 518, 15–29.

Rohlfis, C., 2023b. Problem-dependent attention and effort in neural networks with applications to image resolution and model selection. Image and Vision Computing 135, 104696.

Roth, M., Anthonio, T., Sauer, A., 2022. SemEval-2022 task 7: identifying plausible clarifications of implicit and underspecified phrases in instructional texts. International Workshop on Semantic Evaluation (SemEval) 16, 1039–49.

Rubin, D.B., 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. Journal of Educational Psychology 66, 688–701.

Russakovsky, O., Deng, J., Su, J., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2015. ImageNet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–52.

Russell, S., Norvig, P., 2020. Artificial intelligence: a modern approach, 4th edition. Pearson, London.

Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R., 2016a. Progressive neural networks. arXiv preprint DOI https://arxiv.org/pdf/1606.04671.

Rusu, A.A., Večerík, M., Rothörl, T., Heess, N., Pascanu, R., Hadsell, R., 2016b. Sim-to-real robot learning from pixels with progressive nets. arXiv preprint DOI https://arxiv.org/pdf/1610.04286v2.

S. J. Kielbow, Daunizeau, J., Friston, K.J., 2008. Perception and hierarchical dynamics. Frontiers in Neuroinformatics 3, 20.

Sadéh, S., Clopath, C., 2021. Inhibitory stabilization and cortical computation. Nature Reviews: Neuroscience 22, 21–37.

Saenko, K., Kulis, B., Fritz, M., Darrell, T., 2010. Adapting visual category models to new domains. European Conference on Computer Vision , 213–26.

Savin, C., Triesch, J., 2014. Emergence of task-dependent representations in working memory circuits. Frontiers in Computational Neuroscience 8.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N.R., Kalchbrenner, N., Goyal, A., Bengio, Y., 2021. Towards causal representation learning. Proceedings of the IEEE 109, 612–34. doi:10.1109/JPROC.2021.3088369.

Schultz, W., 1998. Predictive reward signal of dopamine neurons. Journal of Neurophysiology 80, 1–27.

Schultz, W., 2007. Multiple dopamine functions at different time courses. Annual Review of Neuroscience 30, 259–88.

Schultz, W., Dayan, P., Montague, P.R., 1997. A neural substrate of prediction and reward. Science 275, 1593–9.

Schwarz, J., Lukejina, T., Czarneeki, E.M., Grabka-Barwinska, A., Teh, Y.W., Pascanu, R., Hadsell, R., 2018. Progress & compress: a scalable framework for continual learning. Proceedings of Machine Learning Research (PMLR) 80.

Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D., 2017. Grad-cam: visual explanations from deep networks via gradient-based localization. IEEE International Conference on Computer vision.

Seung, H.S., 2018a. Two “correlation games” for a nonlinear network with Hebbian excitatory neurons and anti-Hebbian inhibitory neurons. arXiv preprint DOI https://arxiv.org/abs/1812.11937.

Seung, H.S., 2018b. Unsupervised learning by a nonlinear network with Hebbian excitatory and anti-Hebbian inhibitory neurons. arXiv preprint DOI https://arxiv.org/abs/1812.11581.

Shalaman, D., Korman, S., 2022. The self-optimal-transport feature transform. arXiv preprint https://arxiv.org/pdf/2204.03065v1.pdf.

Shalit, U., Johansson, F.D., Sontag, D., 2017. Estimating individual treatment effect: generalization bounds and algorithms. International Conference on Machine Learning 70, 3076–85.

Shang, J., Wang, S., Sun, Y., Yu, Y., Zhou, Y., Xiang, L., Yang, G., 2022. X-PuDu at SemEval-2022 task 7: a replaced token detection task pre-trained model with pattern-aware ensembling for identifying plausible clarifications. International Workshop on Semantic Evaluation (SemEval) 16, 1078–83.

Shankar, S., Piratla, V., Chakrabarti, S., Chaudhuri, S., Jyothi, P., Sarawagi, S., 2018. Generalizing across domains via cross-gradient training. International Conference on Learning Representations .

Shapley, L.S., 1953. A value for n-person games, in: Kuhn, H.W., Tucker, A.W. (Eds.), Contributions to the theory of games (AM-28), volume II. Princeton University Press, Princeton, NJ, pp. 307–17.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on image data augmentation for deep learning. Journal of Big Data 6, 60.

Shu, X., Qi, G., Tang, J., Wang, J., 2015. Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation. ACM Multimedia Conference (MM) 23, 35–44.

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., 2016. Mastering the game of go with deep neural networks and tree search. Nature 529, 484–9.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., Driessche, G., Graepel, T., Hassabis, D., 2017. Mastering the game of go without human knowledge. Nature 550, 354–9.

Simon, H.A., Newell, A., 1958. Heuristic problem solving: the next advance in operations research. Operations Research 6, 1–10.

Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations .

Simons, S.H., 2008. Heuristic problemsolving:thefinaladvance. Nature 435, 259–88.

Snel, J., Swersky, K., Zemel, R., 2017. Prototypical networks for few-shot learning. Neural Information Processing Systems 31, 4080–90.

Soyan, P., Ionescu, R.T., Rota, P., Sebe, N., 2022. Curriculum learning: a framework for continual learning. Proceedings of Machine Learning Research (PMLR) 80.

Speck, R., Chin, J., Hrvasic, C., 2017. Conceptnet 5.5: an open multilingual graph of general knowledge. Proceedings of AAAI 31.

Speck, R., Hrvasic, C., 2012. Representation general relational knowledge in conceptnet 5. Conference on Language Resources and Evaluation (LREC), 3679–86.
Xu, X., Liu, L., Li, B., 2020. A survey of CAPTCHA technologies to distinguish between human and computer. Neurocomputing 408, 292–307.

Yang, J., Zhou, K., Li, Y., Liu, Z., 2021. Generalized out-of-distribution detection: a survey. arXiv preprint https://arxiv.org/abs/2110.11334.

Yang, K., Wu, Y., Kirillov, A., Massa, F., Girshick, R., Dollar, P., 2020. fvcore: A collection of core functionality for computer vision libraries. https://github.com/facebookresearch/fvcore.

Yao, X., Bai, Y., Zhang, X., Zhang, Y., Sun, Q., Chen, R., Li, R., Yu, B., 2022. PCL: proxy-based contrastive learning for domain generalization. Computer Vision and Pattern Recognition, 7097–107.

Yu, J., Wang, Z., V., Yeung, L., Seyedhosseini, M., Wu, Y., 2022. Coca: contrastive captioners are image-text foundation models. Transactions on Machine Learning Research.

Yuan, L., Li, H., Xia, B., Gao, C., Liu, M., Yuan, W., You, X., 2022. Recent advances in concept drift adaptation methods for deep learning. International Joint Conference on Artificial Intelligence 31, 2805–24.

Zador, A.M., 2019. A critique of pure learning and what artificial neural networks can learn from animal brains. Nature Communications 10, 3770.

Zagoruyko, S., Komodakis, N., 2017. Wide residual networks. arXiv preprint https://arxiv.org/abs/1605.07146.

Zanga, A., Ozkirimli, E., Stella, F., 2022. A survey on causal discovery: theory and practice. International Journal of Approximate Reasoning 151, 101–29.

Zenke, F., Poole, B., Ganguli, S., 2017. Continual learning through synaptic intelligence. Proceedings of Machine Learning Research (PMLR) 70.

Zhan, Z., Li, J., Zhang, J., 2022. Evolutionary deep learning: a survey. Neurocomputing 483, 42–58.

Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2017. Understanding deep learning requires rethinking generalization. International Conference on Learning Representations.

Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O., 2021. Understanding deep learning (still) requires rethinking generalization. Communications of the ACM 64, 107–15.

Zhang, J., Berridge, K.C., Tindell, A.J., Smith, K.S., Aldridge, J.W., 2009. A neural computational model of incentive salience. PLos Computational Biology 5.

Zhang, Y., Gong, K., Ding, X., Zhang, K., Lv, F., Keutzer, K., Yue, X., 2023. Towards unified and effective domain generalization. arXiv preprint https://arxiv.org/abs/2310.10068.

Zhang, Y., Yang, Q., 2022. A survey on multi-task learning. IEEE Transactions on Knowledge and Data Engineering 34, 5586–609.

Zhou, K., Liu, Z., Qiao, Y., Xiang, T., Loy, C.C., 2022. Domain generalization: a survey. IEEE Transactions on Pattern Recognition 45(4), 4396–415.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q., 2021. A comprehensive survey on transfer learning. Proceedings of the IEEE 109(1), 43–76.