



**Figure 6.13** Schematic illustration of transfer learning. (a) A network is first trained on a task with abundant data, such as object classification of natural images. (b) The early layers of the network (shown in red) are copied from the first task and the final few layers of the network (shown in blue) are then retrained on a new task such as skin lesion classification for which training data is more scarce.

ited number of iterations to ensure that the network does not over-fit to the relatively small data set available for the new task.

A related approach is *multitask learning* (Caruana, 1997) in which a network jointly learns more than one related task at the same time. For example, we might wish to construct a spam email filter that allows different users to have different classifiers tuned to their particular preferences. The training data may comprise examples of spam email and non-spam email for many different users, but the number of examples for any one user may be quite limited, and therefore training a separate classifier for each user would give poor results. Instead, we can combine the data sets to train a single larger network that might, for example, share early layers but have separate learnable parameters for the different users in later layers. Sharing data across tasks allows the network to exploit commonalities amongst the tasks, thereby improving the accuracy for all users. With a large number of training examples, a deeper network with more parameters can be used, again leading to improved performance.

Learning across multiple tasks can be extended to *meta-learning*, which is also called *learning to learn*. Whereas multitask learning aims to make predictions for a fixed set of tasks, the aim of meta-learning is to make predictions for future tasks that were not seen during training. This can be done by not only learning a shared