Deep Learning

Yuan An

In the last article, supervised learning and backpropagation are introduced in Lecun *et al*. review paper [1]. And convolutional neural networks (ConvNets) and image understanding with deep convolutional networks will be talked about in detail.

ConvNets are designed to process data composed of multiple arrays. *E.g*. a color image composed of thre 2D arrays containing pixel intensities in the three color channels (RGB). And many data modalities are in the form of multiple array: 1D for signals and sequences; 2D for images or audio spectrograms; 3D for video or volumetric images. And the four key ideas behind ConvNets that take advantage of the properties of natural signals are local connections, shared weights, pooling and the use of many layers.

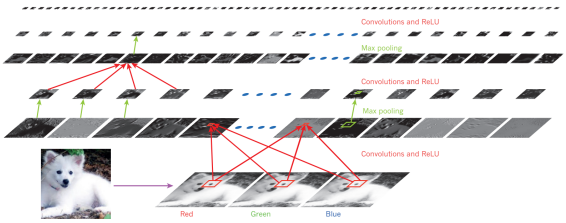

Figure 1. Inside a convolutional network. The output (not the filters) of each layer (horizontally) of a typical convolutional network architecture applied to the image of a Samoyed dog (bottom left; and RGB (red, green, blue) inputs, bottom right). Each rectangular image is a feature map corresponding to the output for one of the learned features, detected at each of the image positions. Information flows bottom up, with lower-level features acting as oriented edge detectors, and a score is computed for each image class in output. ReLU, rectified linear unit.

The architecture of a typical ConvNet is shown in Figure 1. It is structured as a series of stages. The first few stage are composed of convolutional layers and pooling layers. And units in a convolutional layers are organized in feature maps, within which each unit is connected to local patches in the feature maps of the previous layer through a set of weights called a filter bank. The result of this local weighted sum is then passed through a non-linearity such as a ReLU. All units in a feature map share the same filter bank. Different feature maps in a layer use different filter banks.

Although the role of the convolutional layer is to detect local conjunctions of features from the previous layer, the role of the pooling layer is to merge semantically similar features into one. Because the relative positions of the features forming a motif is somewhat different, reliably detecting the motif can be done by coarse-graining the position of each feature. A typical pooling unit computes the maximum of a local patch of unites in one feature map (or in a few feature maps). Neighboring pooling units take input from patches that are shifted by more than one row or column, thereby reducing the dimension of the representation and creating an invariance to small shifts and distortions.

Deep neural networks exploit the property that many natural signals are compositional hierarchies, in which higher-level features are obtained by composing lower-level ones. In images, local combinations of edges form motifs, motifs assemble into parts, and parts form objects. Similar hierarchies exist in speech and text from sounds to phones, phonemes, syllables, words and sentences. The pooling allows representations to vary very little when elements in the previous layer vary in position and appearance.

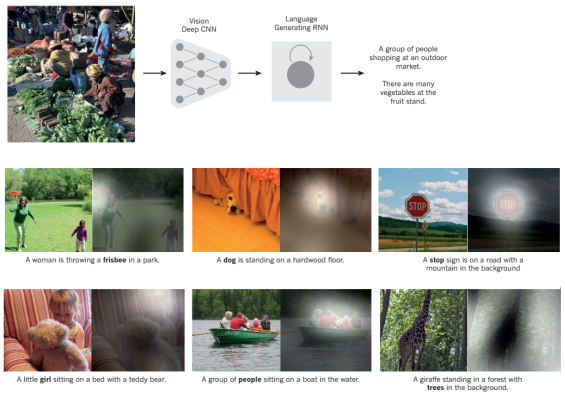

Figure 2. From image to text. Captions generated by a recurrent neural network (RNN) taking, as extra input, the representation extracted by a deep convolution neural netowrk (CNN) from a test image, with the RNN trained to 'translate' high-level representations of images into captions (top). When the RNN is given the ability to focus its attention on a different location in the input image (middle and bottom; the lighter patches were given more attention) as it generates each word (bold), we found that it exploits this to achieve better 'translation' of images into captions.

Since the early 1990s, there are numerous applications of ConvNets, starting with time-delay neural networks for

speech recognition [5] and document reading [2]. Then a number of ConvNet-based optical character recognition and handwriting recognition systems were later deployed by Microsoft [4]. ConvNets were also experimented with in the early 1990s for object detection in natural images, including face and hands.

Since the early 2000s, ConvNets have been applied with great success to the detection, segmentation and recognition of object and regions in images. There were all tasks in which labeled data was relatively abundant, such as traffic sign recognition, the segmentation of biological images and the detection of faces, text, pedestrians and human bodies in natural images. A major recent practical success of ConvNets is face recognition [3].

Despite these successes, ConvNets were largely forsaken by the mainstream computer version and machine learning communities until the ImageNet competition in 2012. When deep convolutional networks were applied to a data set of about a million images from the web that contained 1,000 different classes, they achieved spectacular results, almost halving the error rates of the best competing approaches. A recent stunning demonstration combines ConvNets and recurrent net modules for the generation of image captions (see Figure 2).

References

[1] Y. Lecun, Y. Bengio, and G. Hinton. *Nature*, 521(7553):436–444, 2015. 1

[2] Y. Lécun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[3] Y. Taigman, M. Yang, Marc, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, pages 1701–1708, 2014. 2

[4] A. T. Visual, P. Y. Simard, D. Steinkraus, and J. C. Platt. Best practices for convolutional neural networks. In *ICDAR*, pages 958–962, 2003. 2

[5] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang. Phoneme recognition using time-delay neural networks. In *IJCNN*, pages 235–241, 1995. 2