Deep Learning

Yuan An

In the last article, convolutional neural networks and image understanding with deep convolutional networks have been introduced. In the rest part of the paper of Lecun *et al*. [5], distributed representations and language processing and recurrent neural networks will be talked about.

Distributed representations and language processing

Deep learning theory shows that deep nets have two different exponential advantages over classic learning algorithm that do not use distributed representations [2]. Both of these advantages arise from the power of composition and depend on the underlying data-generating distribution having an appropriate componential structure [1]. First, learning distributed representations enable generalization to new combination of the values of learned features beyond those seen during training. Second, composing layers of representation in a deep net brings the potential for another exponential advantage.

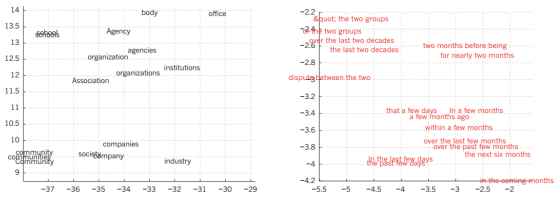

Figure 1. Visualizing the learned word vectors. On the left is an illustration of word representations learned for modeling language, non-linearly projected to 2D for visualization using the t-SNE algorithm. On the right is a 2D representation of phrases learned by an English-to-French encoder-decoder recurrent neural network.

The hidden layers of a multi-layer neural network learn to represent the input data for predicting target output more easily. There is a good example of training a multi-layer neural network to predict the next word in a sequence from a local context of earlier words [3]. Each word in the context is presented to the network as a one-of-N vector. That is to say, one component has a value of 1 and the the rest are 0. In the first layer, each word creates a different pattern of activations, or word vectors (see Figure 1).

When trained to predict the next word in a new story, for example, the learned word vectors for Tuesday and Wednesday are very similar, as are the word vectors for Sweden and Norway. Such representations are called distributed representations because their elements (the features) are not mutually exclusive and their many configurations correspond to the variations seen in the observed data. These word vectors are composed of learned features that were not determined ahead of time by experts, but automatically discovered by the neural network. Vector representations of words learned from text are now very widely used in natural language applications.

Recurrent neural networks

When it comes to back-propagation firstly, the most exciting use was for training recurrent neural networks (RNNs). For tasks involved in sequential inputs, such as speech and language, researchers prefer use RNNs (see Figure 2). RNNs process an input sequence one element at a time, maintaining in their hidden units a 'state vector' that implicitly contains information about the history of all the past elements of the sequence. When we consider the outputs of the hidden units at different discrete time steps as if they were the outputs of different neurons in a deep multi-layer network (Figure 2, right), it becomes clear how we can apply backpropagation to train RNNs.

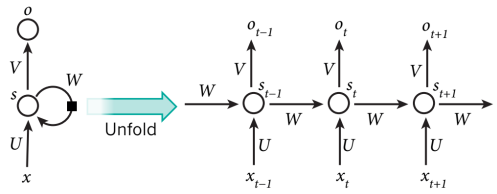

Figure 2. A recurrent neural network and the unfolding in time of the computation involved in its forward computation. The artificial neurons get inputs from other neurons at previous time steps (this is represented with the black square, representing a delay of one time step, on the left).

RNNs are very powerful dynamic systems, but training them has proved to be problematic because the back-propagated gradients either grow or shrink at each time step, so over many time steps they typically explode or vanish.

Owing to advances in their architecture and ways of training them, RNNs have been found to be very good at pre-

dicting the next character in the text [7] or the next word in a sequence [6], but they can also be used for more complex tasks.

Instead of translationg the meaning of a French sentence into an English sentence, one can learn to 'translate' the meaning of an image into an English sentence.

RNNs, once unfolded in time, can be seen as very deep feedforward networks in which all the layers share the same weights. Although their main purpose is to learn long-term dependencies, theoretical and empirical evidence shows that it is difficult to learn to store information for very long.

To solve this problem, one idea is to augment the network with an explicit memory. The first proposal of this kind is the long short-term memory (LSTM) networks that use special hidden units, the natural behaviour of which is to remember inputs for a long time.

LSTM networks have subsequently proved to be more effective than conventional RNNs, especially when they have several layers for each time step [4], enabling an entire speech recognition system that goes all the way from acoustics to the sequence of characters in the transcription.

The future of deep learning

Unsupervised learning had a catalytic effect in reviving interest in deep learning, but has since been overshadowed by the successes of purely supervised learning.

Natural language understanding is another area in which deep learning is poised to make a large impact over the next few years.

References

[1] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE TPAMI*, 35(8):1798–1828, 2013. 1

[2] Y. Bengio, O. Delalleau, and N. L. Roux. The curse of highly variable functions for local kernel machines. In *NIPS*, pages 107–114, 2005. 1

[3] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. In *NIPS*, pages 1137–1155, 2003. 1

[4] A. Graves, A. R. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *ICASSP*, pages 6645–6649, 2013. 2

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

[6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013. 2

[7] I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. In *ICML*, pages 1017–1024, 2016. 2