History of Recurrent Neural Networks and Deep Neural Networks Over Time

Hacer Çoban

Deniz Küçükahmetler

Zeynep Yaradanakul

Zeynep Bekar

Mert Malaz

Sabancı University

PSY 350: Introduction to Neuroscience

Nihan Alp

January 22, 2021

Abstract

This review presents a brief history of deep neural networks (DNN) and recurrent neural networks (RNN) over time to display their improvement in terms of functionality and their application fields. The review starts with a brief introduction to the DNN and RNN field, including their fundamental working principles, the background of the neural network (NN) field, the timeline of their usage, and areas of their implementations. In the following pages of the review, the NN concept is presented, DNN and RNN mechanisms are further investigated, their relationship with the biological field is also pointed out and some contributions are included to show the recent developments in the field.

Keywords: neural network, deep neural network, recurrent neural network, layer, neuron, feedback mechanism, feedforward mechanism

Introduction

NNs are computing systems consisting of artificial neurons influenced by biological networks of the brain, in which the algorithm analyzes training examples to perform its tasks (Samek et al., 2017). Neurons are the units of the brain and the nervous system, in which they receive sensory input from the external environment, interpret those inputs, and send motor commands. Neurons work together through electrical and chemical signals, constructing a perfect decision-making mechanism and NNs are influenced by this architecture of the human brain. The functionality of neurons is reflected to nodes of the NNs; taking input and producing output and these nodes are organized into layers to comprise the NN (Samek et al., 2017). However, the difference is that NNs use mathematical functions for their communication. There are several types of NNs and their common components are neurons, synapses, weights, functions, and biases in general. DNN and RNN are the most preferred variants of NNs in which the inputs are processed with a specified number of layers (Samek et al., 2017).

RNN and DNN both contribute to the understanding of the brain and approach real-world problems with NN solutions. They are both inspired by the biological principles of the brain. There are many common characteristics they have such as consisting of “neuron” layers and having communication between layers to come up with a solution to a problem. However, they follow different approaches for processing the data. RNN has a feedback mechanism to recurrently change the network whereas DNN does not have any storage for processed inputs. That’s why DNN does not have a recurrent behavior. Each day, it is becoming more popular because it works efficiently and accurately.

The development of the first NNs starts with the intention of modeling how neurons in the brain might work and people from various areas such as mathematics, neurophysiology, and psychology involved in the process. In 1943, the first implementation was done using electrical circuits to model NNs (McCulloch & Pitts, 1943). Later on, this fresh area caught the attention of many other scientists thus lots of new research took place. Initial practical uses of NN start with filtering noises from analog signals, like signals that a phone line receives, by showing the network which signals are possible noises (Widrow & Hoff, 1989). At that time, the primitiveness of the system prevented researchers from creating different uses and required working on improvements to the system rather than creating solutions to real-life problems. The golden age of NNs started in the 1990's as the foundations of complex and efficient systems such as RNN, DNN were implemented. Despite the success, the improvements were bottlenecked by the hardware of that era, and fruits of those works were received later in the 2000s as the speed of hardware became compatible with the needs of the NNs systems. NNs are spreading to more fields every day while maintaining their importance.

Today, NNs are used in many different fields. It can be categorized into some of the main areas. In computer science, it is used in natural language processing: machine translation, speech recognition, grammar learning, speech synthesis (Hannun et al., 2014). In health: anomaly detection, protein homology detection, subcellular localization of the proteins, drug discovery, and toxicology. Also, in image recognition, robot control, visual art processing, human action recognition, music composition, rhythm learning (Schmidhuber, 2015). The aim of this study is to observe the history of DNN and RNN over time, and their implementation fields.

Neural Networks

NN is a computational model that is based on the functions of biological networks and it can be used to extract patterns to detect complex correlations of instances (Sonali & Wankar, 2014). A trained neural network can be conceptualized as an expert in the analysis of the given information. It is developed through the decades with increasing efficiency, therefore NNs have a notable historical background.

The first paper about the base theorems of NNs was written in 1943 (McCulloch & Pitts, 1943). McCulloch and Pitts mathematically explained the way of conceptualizing the brain as a net that consists of neurons by coming up with the main characteristics of the network such as neuronal states, connections of the neurons, and laws of the neuron activities (McCulloch & Pitts, 1943). RNN's and DNN's nodes are similar to neurons, their data processing behavior can be seen as inherited processing of the brain (McCulloch & Pitts, 1943). Besides, the connectivity between nodes in NNs is also inspired by the brain. From 1943 to this date, scientists have tried to make the functioning of the NN more similar to the functioning of the brain (Federico, 2009). As time goes by, NNs have been found more and more similar to neurobiological systems. The brain has a large number of parameters while processing information, so do NNs (Richards et al., 2019). This makes it easier to understand the brain and predict the output (Richards et al., 2019). Part of the biological data known today can be explained by NN models (Richards et al., 2019). More specifically, NN models mimic the primate perceptual system in some conditions (Richards et al., 2019). Moreover, deep learning algorithms used in NNs explain some neurobiological concepts such as "grid cells, shape tuning, temporal receptive fields, visual illusions" (Richards et al., 2019). Therefore, we know that NNs emerged with neurobiological truths and now it has a high potential of contributing to neuroscience (Richards et al., 2019).

There are lots of contributions to science. Firstly, the emotion recognition model was created by using the features of facial expression in neural network models such as amusement, anger, disgust, fear, and sadness. In order to distinguish these 5 emotions, a classification process is performed by detecting the face in the videos with geometric data-based features. Surprisingly, even the distinction between anger and disgust emotions is nearly 82% accurate (Mostafa, Khalil, and Abbas 2019). Additionally, emotion labels were created with most of the sections, with the usage of a database containing spontaneous emotional conversations consisting of 12-hour audio-visual recordings of a German TV program. These emotion labels are divided into 3 principles: balance, activation, and dominance. Such data can be used to recognize emotion in both speech and face recognition with spontaneous speech analysis (Grimm, Kroschel, and Narayanan, 2008).

Another contribution is that it also has offline handwriting recognition features in the NN model. Normally, these models are taught a language, and then the model can work through the language it learns. However, the advantage of this model is that it can work for all languages. That is, the model interprets languages itself, without having any idea about the language. A very general model can be created and used in any language. For example, the model achieved 87.2% accuracy even though it does not understand Arabic (Graves, 2012).

Additionally, in 2015, the AlphaGo Zero model emerged with a NN that knows nothing about the game of Go. And then, this model could play games against itself using powerful search algorithms. It beat Go world champions in different places. The performance of the system increases a little more each time. This contributes to increasingly accurate NNs and stronger versions (Silver and Hassabis, 2017).

Overview of Deep Neural Networks

As demonstrated in figure 1, DNNs have multiple hidden layers; an input layer, hidden layers, and an output layer. Each node in the hidden layers increases the effect of input on the output and deep refers to this model's hidden layers being more than one. Having multiple layers increases the accuracy of the model because the information is going to be processed more and the network gains the ability of learning complex patterns. Each calculation in the hidden layer's activation function is a linear operation that includes a matrix and bias which are combined with a parameter. This activation function between nodes operates similar to neurons which makes the NN more biologically plausible. The weights of the nodes represent the importance of the inputs for the output. The input data is taken by the nodes of the first hidden layer, and the first hidden layer is an input for the next hidden layer. This communication goes on. In the end, the outcome is produced by all of these nodes' communication via their activation functions (Samek et al., 2017). For example, the outcome might be a prediction step in which the algorithm decides which class the input belongs to and the output is determined by probability comparisons with a threshold.

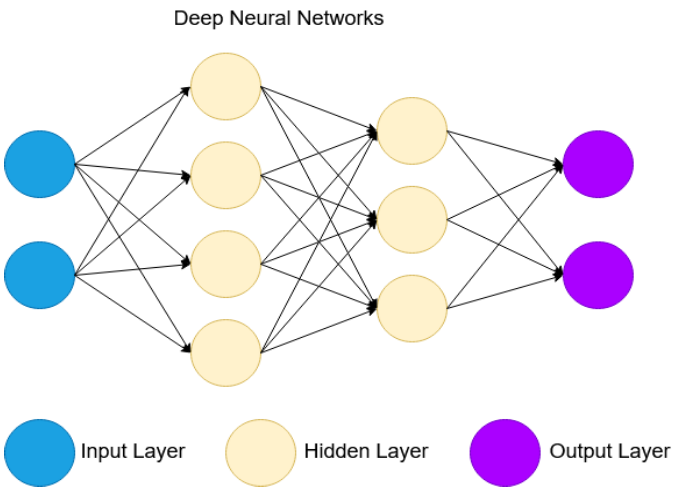

Figure 1: DNN Structure (De Mulder et al., 2019). Redrawn for clarity.

DNN is inspired by the physiology of the human brain and its biological network structure with the existence of many simple processing units that are connected to each other, quite similar to the function and organization of neurons (Cichy & Kaiser, 2019). Throughout the input, output, and hidden layers, DNN uses a hierarchical architecture, similar to the brain's hierarchical functioning while carrying out cognitive tasks (Cios, 2018). Similarly, communication between nodes is much alike to the communication between neurons via neurotransmitters by either increasing or decreasing the likelihood for a response.

In 1943, the foundations of NNs were founded by McCulloch and Pitts by showing that the neural activities can be denoted with the terms of propositional logic. Especially, the "all-or-none" property of the neural activity enabled them to arrive at such a conclusion (McCulloch & Pitts, 1943). After 15 years, Rosenblatt invented the perceptron algorithm which is responsible for the supervised learning of binary classifiers (Rosenblatt, 1958). By this invention, he showed that a system that consists of randomly connected units can learn to create a link between specific responses and specific stimuli concerning some parametric constraints (Alom et al., 2018). However, the limitations of the perceptron algorithm were demonstrated in 1969. Therefore, the research in this area was ceased until 1985 (Alom et al., 2018). Nevertheless, the area was revived by the backpropagation algorithm which was used for training feedforward NNs (Alom et al., 2018). Three years after this revival in the field, Fukushima proposes neocognitron, a multi-layered ANN (artificial neural network) that works hierarchically for visual pattern recognition (Alom et al., 2018; Fukushima, 2011).

Overview of Recurrent Neural Networks

RNNs, similar to DNNs, consist of multiple layers; an input layer, multiple hidden layers, and an output layer as seen in figure 2. However, unlike the DNN, nodes inside the hidden layers of RNN are capable of storing previously processed data, it doesn't store all the previous data but only the latest previous output, then it is given as an input to the current function. The preservation of the previous output and combining it with the next input is achieved by creating a chain structure with a loop for each hidden layer node as shown in figure 2. This allows hidden layers to use the output of the previous timestamp as an input to the current timestamp. Since the processing of previous and current inputs are done in only one node, the weight will be shared across all the timestamps for a node. However, the weight of the previous timestamp is a hyperparameter that can be tuned to obtain better results. In this fashion, function at the latest timestamp will create a cumulative output for each node and transfer it to the next layer, then the same process will be repeated until the output layer is reached. This chain structure is perfect for processing sequential data that requires information from the past to interpret the current input like text and speech recognition (Graves & Jaitly, 2014), time-stamped data such as recognition and prediction of human body pose (Fragkiadaki et al., 2015). RNN suffers from the excess amount of computation needed due to the backward propagation through time, thus training an RNN model is significantly slower than models that do not require remembering previous outputs (Yu et al., 2021). In the RNN model, each neuronal input processes a non-linear output with respect to the integrated input. This can be useful to understand the real-world stimuli of the brain.

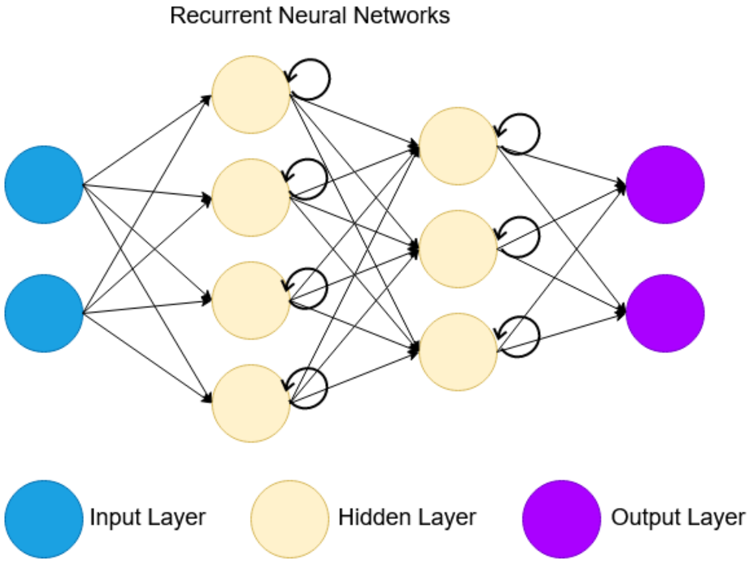

Figure 2: RNN Structure (De Mulder et al., 2019). Redrawn for clarity.

RNN has strong connections with biological NNs in terms of long-term dependencies (Güçlü & van Gerven, 2017). Neurons send information in every direction in the brain, to various distances. RNN behaves similarly but it does not support a detailed classification of neurons. The feed-back connectivity of RNN can be seen in several functions of biological networks such as iterative sensory processing, short-term memory, top-down attention, long-term memory (Hochreiter & Schmidhuber, 1997b; Mittal et al., 2020; Pitti et al., 2017).

In 1982, John Hopfield discovered Hopfield Networks which includes the basic properties of RNN and is perceived as a pioneer for further improvements in RNN. Hopfield Network contains stationary inputs (i.e. it doesn't process sequences of inputs) that is why it is not counted as a complete RNN (Sathasivam & Wan Abdullah, 2008). Then, Elman Network was founded in 1990 (Elman, 1990). It proposed the idea of "networks with memory" which is an important part of RNNs (Song, 2010). In 1997, Bengio established the gradient descent/vanishing problem for RNN's. This problem is explained as a failure of the learning process while the activation of neurons within the network are so small that the parameter updates are no longer effective which causes the recurrent network to stop in some cases. (Hochreiter & Munchen, 1998). The vanishing gradient problem was solved with long-short term memory (LSTM) networks in the same year by Hochreiter (Hochreiter & Schmidhuber, 1997). LSTMs form a part of RNN today. One year later, Schuster came up with bidirectional RNNs (BRNN) (Schuster & Paliwal, 1997). BRNNs overcome the problem of getting a present input to produce an output by taking inputs from opposite directions in time simultaneously. Therefore, the past and future states are available for getting information from the current (present) state. BRNNs can be combined with LSTM for several applications such as speech recognition.

Until 2000, the basics of RNN were completed. Further developments for RNN mostly improved the already existing algorithms and extended the usages in different applications. In 2000, Felix Gers showed a weakness in LSTM and developed an algorithm for it (Gers et al., 2000). The problem was keeping track of very long sequences. The protocol of processing the data recurrently makes LSTM stronger. However, if the processed data is not necessary to find the accurate output, it means that it is filling space in the memory unnecessarily especially when the input sequence is too long. To overcome this problem, he made a “forgetting” mechanism to delete the unnecessary data in the memory as LSTM processes the inputs recurrently. It is observed that LSTM outperformed many RNN and LSTM algorithms (Gers et al., 2000). Alex Graves made numerous improvements in this field. Between 2005-2009, he brought a new perspective to RNN. In 2005, Graves developed bidirectional LSTM (BLSTM) by combining the earlier discussed bidirectional property of BRNN with LSTM. He showed that LSTM is faster and more accurate than RNN. Furthermore, BLSTM is better than regular (unidirectional) LSTM for doing classification (Graves et al., 2005). He improved the usage of RNN for hand-writing recognition and speech recognition. In 2009, the RNN based algorithm won the hand-writing recognition contest by outperforming the traditional algorithms for the first time (Grosicki & El Abed, 2009). Afterwards, Graves suggested a two-dimensional usage of LSTM. Until then, LSTM was only used for one-dimensional inputs such as speech. By establishing multi-dimensional LSTM, RNN gained broader usage. It made it possible to use RNN for computer vision, video processing, and other applications (Graves et al., 2007). In 2010, Mikolov developed RNN based language model and improved speech recognition with RNN (Mikolov et al., 2010). Mikolov with his group created a backpropagation through time (BPTT) algorithm to improve the previously stated language model and to make wider use of RNN for time series data (Deng et al., 2018). Afterwards, in 2015, Mikolov developed a structurally constrained recurrent network (SCRN) which performs better for language modeling (Project & Olzhas, 2017). After those progresses through the years, speech recognition and analysis were improved a lot which resulted in wider usage of RNN in companies. Google announced that they officially started using RNN in Android devices for voice search which requires speech recognition and speech to text conversion with the evolution of RNN.

As time went by, RNN entered more fields and its usage got spread out in an interdisciplinary manner. By combining different properties of RNN and different NN algorithms such as CNN, it is improved more and more as discussed above (Zhang et al., 2018). There are innumerable other algorithms that have been done and they are still developing every day.

Conclusion

DNN and RNN are the most popular versions of NNs and they are being improved over time. Neurons' working environment through both chemical and electronic signals create a perfect decision-making mechanism which is reflected in these NNs. DNN and RNN mainly differentiate in the sense of having extra storage for having an elaborate feedback mechanism. RNN’s recurrent feedback mechanism requires storage for the processed input throughout the operation. However, DNN does not have extra storage for that but many hidden (deep) layers. That is why DNN does not have an advanced feedback mechanism as RNN does. They both learn complex patterns and produce deterministic outputs. They are both open to development in terms of improving their existing algorithms and combining them to implement new algorithms. As discussed before, previously done studies showed numerous algorithmic approaches for both RNN and DNN which serve different purposes. For instance, one algorithm can serve better for speech recognition whereas the other can serve better in games as it is seen in history. Deeper observation in neuroscience is required in order to come up with better neural network algorithms. As the similarity between neurons and algorithm nodes increase, better decision-making mechanisms can be implemented. Consequently, NNs are inspired by biological brain mechanisms and they are being developed by the contributions of the field of neuroscience.

References

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., van Essen, B. C., Awwal, A. A. S., & Asari, V. K. (2018). The history began from AlexNet: A comprehensive survey on deep learning approaches. *ArXiv*.

Cichy, R. M., & Kaiser, D. (2019). Deep Neural Networks as Scientific Models. *Trends in Cognitive Sciences*, *23*(4), 305–317.

Cios, K. J. (2018). Deep neural networks—A brief history. *Studies in Computational Intelligence*, *738*, 183–200.

De Mulder, W., Bethard, S., & Moens, M.-F. (2015). A survey on the application of recurrent neural networks to statistical language modeling. *Computer Speech & Language*, *30*(1), 61–98.

Deng, H., Zhang, L., & Shu, X. (2018). Feature memory-based deep recurrent neural network for language modeling. *Applied Soft Computing Journal*, *68*, 432–446.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*.

Federico, Marini (2009). 3.14 - Neural Networks. In *Comprehensive Chemometrics*.

Fragkiadaki, K., Levine, S., Felsen, P., & Malik, J. (2015). Recurrent network models for human dynamics. *Proceedings of the IEEE International Conference on Computer Vision*.

Fukushima, K. (2011). Increasing robustness against background noise: Visual pattern recognition by a neocognitron. *Neural Networks*, *24*(7), 767–778.

Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, *12*(10), 2451–2471.

Graves, A., & Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. *31st International Conference on Machine Learning, ICML 2014*.

Graves, A., Fernández, S., & Schmidhuber, J. (2005). Bidirectional LSTM networks for improved phoneme classification and recognition. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *3697 LNCS*, 799–804.

Graves, A., Fernández, S., & Schmidhuber, J. (2007). Multi-dimensional recurrent neural networks. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *4668 LNCS*(PART 1), 549–558.

Graves, Alex. 2012. “Offline Arabic Handwriting Recognition with Multidimensional Recurrent Neural Networks.” in *Guide to OCR for Arabic Scripts*.

Grimm, Michael, Kristian Kroschel, and Shrikanth Narayanan. 2008. “The Vera Am Mittag German Audio-Visual Emotional Speech Database.” in *2008 IEEE International Conference on Multimedia and Expo, ICME 2008 - Proceedings*.

Grosicki, E., & El Abed, H. (2009). ICDAR 2009 handwriting recognition competition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, 1398–1402.

Güçlü, U., & van Gerven, M. A. J. (2017). Modeling the dynamics of human brain activity with recurrent neural networks. *Frontiers in Computational Neuroscience*.

Hochreiter, S., & Munchen, T. U. (1998). the Vanishing Gradient Problem During Learning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *2*, 107–116. http://www.bioinf.jku.at/publications/older/2304.pdf

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*.

Mikolov, T., Karafiát, M., Burget, L., Jan, C., & Khudanpur, S. (2010). Recurrent neural network based language model. *Proceedings of the 11th Annual Conference of the International Speech Communication Association, INTERSPEECH 2010*, *September*, 1045–1048.

Mittal, S., Lamb, A., Goyal, A., Voleti, V., Shanahan, M., Lajoie, G., Mozer, M., & Bengio, Y. (2020). Learning to Combine Top-Down and Bottom-Up Signals in Recurrent Neural Networks with Attention over Modules. In *arXiv*.

Mostafa, Amr, Mahmoud I. Khalil, and Hazem Abbas. 2019. “Emotion Recognition by Facial Features Using Recurrent Neural Networks.” in *Proceedings - 2018 13th International Conference on Computer Engineering and Systems, ICCES 2018*.

Pitti, A., Gaussier, P., & Quoy, M. (2017). Iterative free-energy optimization for recurrent neural networks (INFERNO). *PLoS ONE*.

Project, C., & Olzhas, B. (2017). *Initial Explorations on Regularizing the SCRN Model*.

Remaida, Ahmed, Aniss Moumen, Younes El Bouzekri El Idrissi, and Zineb Sabri. 2020. "Handwriting Recognition with Artificial Neural Networks a Decade Literature Review." in *ACM International Conference Proceeding Series*.

Richards, B. A., Lillicrap, T. P., Beaudoin, P., Bengio, Y., Bogacz, R., Christensen, A., Clopath, C., Costa, R. P., de Berker, A., Ganguli, S., Gillon, C. J., Hafner, D., Kepecs, A., Kriegeskorte, N., Latham, P., Lindsay, G. W., Miller, K. D., Naud, R., Pack, C. C., … Kording, K. P. (2019). A deep learning framework for neuroscience. In *Nature Neuroscience*.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408.

Samek, W., Binder, A., Montavon, G., Lapuschkin, S., & Müller, K. R. (2017). Evaluating the visualization of what a deep neural network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(11).

Sathasivam, S., & Wan Abdullah, W. A. T. (2008). Logic Learning in Hopfield Networks. *Modern Applied Science*, *2*(3), 1–8.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2673–2681.

Silver, David, and Demis Hassabis. 2017. "AlphaGo Zero: Starting from Scratch." *DeepMind*.

Sonali, Maind, B., & Wankar, P. (2014). Research Paper on Basic of Artificial Neural Network.

Song, Q. (2010). On the weight convergence of Elman networks. *IEEE Transactions on Neural*

Networks, *21*(3), 463–480.

Widrow, B., & Hoff, M. E. (1989). Adaptive switching circuits. *Wescon Conference Record*, 709–717.

Yu, W., Gonzalez, J., & Li, X. (2021). Fast training of deep LSTM networks with guaranteed stability for nonlinear system modeling. *Neurocomputing*, *422*, 85–94.