Deep Laearning Evolving Machine Intelligence Towards Tomorrow's Intelligent Network Traffic Control System

Zubair Md. Fadlullah,Fengxiao Tang,Bomin Mao,Nei Kato,Osamu Akashi,Takeru Inoue and Kimihiro Mizutani

November 16, 2017

Topics I am going to cover in presentation?

- Introduction
- Few Deep Learning architectures overview
- Deep learning enablers for Network Traffic Control System
- Existing Deep Learning implemented Network Areas
- New Research by Author: Deep Learning Based Routing in brief
- Paper Conclusion

Introduction

- Currently, network traffic control system are mainly composed of the Internet core and wired/wireless heterogeneous backbone networks.
- In recent past years, these packet-switched systems are experiencing an explosive network traffic growth due to rapid development of communication technologies and still growing.
- Deep Learning, with the recent breakthrough in the machine learning/intelligence area, appears to be more viable approach for the network operators to configure and manage their network in a more intelligent and autonomous way.

- In this paper author focused on investigating how Deep learning applications are disrupting the network traffic control systems.
- Talked about a new emerging interdisciplinary area "Network Traffic Control system with Deep Learning".
- The whole paper authors divide into four parts

Overview of Deep Learning Architectures

- Depending on how the Deep Learning architectures are intended for use, they can be broadly categorized into three types, generative, discriminative and hybrid deep architectures.

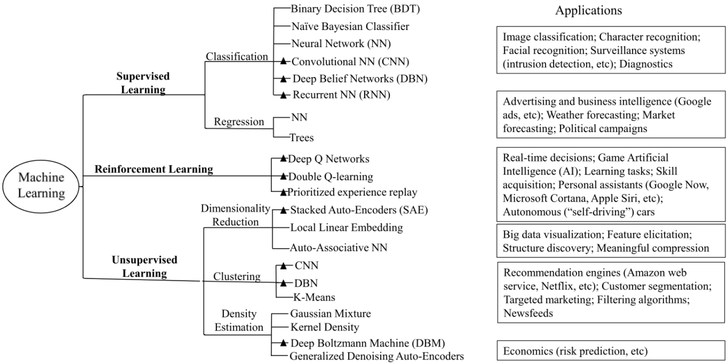
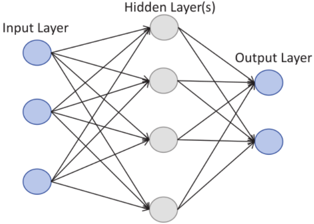


Fig. 2. Various Machine Learning techniques exploited for solving a myriad of computer science problems. It may be noticed that deep learning techniques which are shown with ▲ labels have emerged recently with their use mainly restricted to objects recognition and have not been applied to intelligent network traffic control systems extensively.

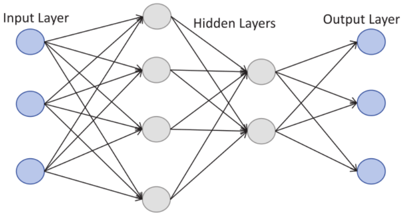
Few Deep Learning Architecture Overview

- CNN -> CNN is a discriminative deep architecture. At the first glance CNN looks quite similar to ANN. Because both the architectures consists of neurons having learnable weights and biases.

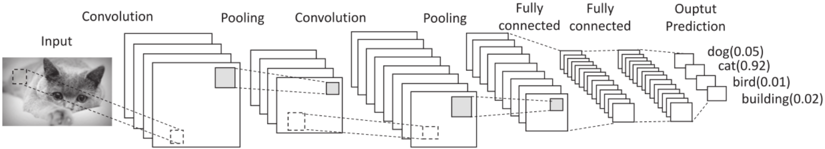
  CNNs have proved to be quite successful in learning task specific features, which have provided much improved results,particularly on different computer vision tasks, in contrast of different ML techniques.



(a) Shallow ANN architecture.



(b) Deep ANN architecture.

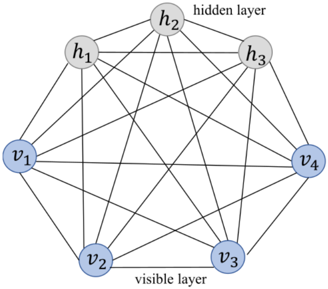


(c) CNN architecture.

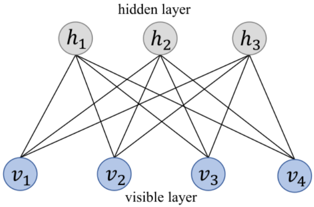
- RNN or LSTM -> The RNN can be considered to be a deep generative architecture. The depth of RNN may be as large as the length of input data sequence. Therefore, the RNN is particularly useful for modeling the sequence data in text and speech. Despite the potential strength, RNN was restricted due to vanishing gradient problem. But recently new optimization method for training RNN evolved using Stochastic Gradient Descent.

- Stacked Auto-Encoder -> An Auto-encoder refers to an ANN aimed to learn efficient coding by encoding a set of data. The encoded data conveys a compressed representation of data set.

- Deep Boltzmann Machine -> A Boltzmann Machine, is a network of binary stochastic units with an "energy" defined for network.



(a) Boltzmann Machine (BM) architecture.



(b) Restricted Boltzmann Machine (RBM) architecture.

- Deep Reinforcement Learning -> Reinforcement learning combines the strength of both supervised and unsupervised learning methods. In a reinforcement learning method, sparse and time-delayed labels (referred to as "rewards") are used based on which the agent has to learn how to behave in a given environment.

  The most well known reinforcement learning technique is Q-learning.

DEEP LEARNING ENABLERS FOR NETWORK TRAFFIC CONTROL SYSTEMS

- GPU-based Software Defined Routers (SDRs)

- GPU-accelerated deep learning platforms:
  1. Tensorflow -> Google's TensorFlow is an open-source interface for accessing the state-of-the-art ML and deep learning algorithms.
  2. Torch -> Based on the Lua programming language, Torch offers several machine/deep learning algorithms with fast and efficient GPU support.

3. Caffe -> The Convolutional Architecture for Fast Feature Embedding, referred to as Caffe, provides the researchers with a robust framework of deep learning algorithms.

4. Deeplearning4j -> Deeplearning4j (DL4J) offers an open source neural network library for Java and Scala programming languages.

5. WILL -> WILL is a high performance deep learning framework, which is supported by C++ and compatible to other interfaces of C, Python and assembly language.

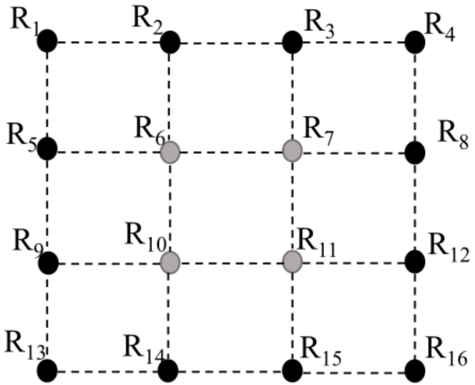
6. CUDA -> Nvidia's CUDA offers a parallel computing platform and programming model based on GPUs. The deep neural network library in CUDA comprises a library of primitives for the standard deep learning routines

APPLICATIONS OF DEEP LEARNING IN NETWORK RELATED AREAS

- Deep Learning in Wireless Sensor Networks
- Network Traffic Classification and Deep Learning
- Network Flow Prediction With Deep Learning
- Deep Learning in Social Networks
- Mobility Prediction With Deep Learning
- Deep Learning in Cognitive Radio and Self-Organized Networks

DEEP LEARNING BASED ROUTING

- Deep learning can be used in a wide range of networking related areas. Since network traffic grows exponentially in the recent years, traffic control is essential to ensure the QoS, especially in the real-time multimedia networks where packet re-transmissions due to the traffic congestion are not a sensible option.
- To solve this shortcoming of conventional routing methods, researchers have explored to adopt machine learning to manage the path intelligently. However, to the best of authors knowledge, contemporary researchers did not exploit deep learning for network traffic control.
- For this research they choose $4 * 4$ wireless mesh backbone network



The considered wireless mesh backbone network.

- In the considered network, assume that the packets are generated only in edge routers and destined for other edge routers since the access terminals are all connected to the edge routers while the inner routers just play the role of forwarding packets.
- Each edge router is assumed to run several Deep Belief Networks(DBNs) to construct the whole paths to other edge routers and attach its packets with the corresponding paths.
- The inner routers do not need to run the DBNs and just read the path to forward the packets.
- For each DBN, the units in its bottom layer are characterized as the traffic patterns of all routers in the network while the top layer represents the next node for an origin-destination pair.

- They adopt a 16-dimensional vector format output to represent the next node path of the DBN such that each of its elements has a binary value.
- Only a single element in this vector can be 1.
- The number of DBNs in the network is 180.
- The training of DBN can be separated into two steps. The first step is pre-training of the architecture with the Greedy Layer-Wise training method, and the following step is to fine-tune the architecture with the back propagation method.
- Error propagation:

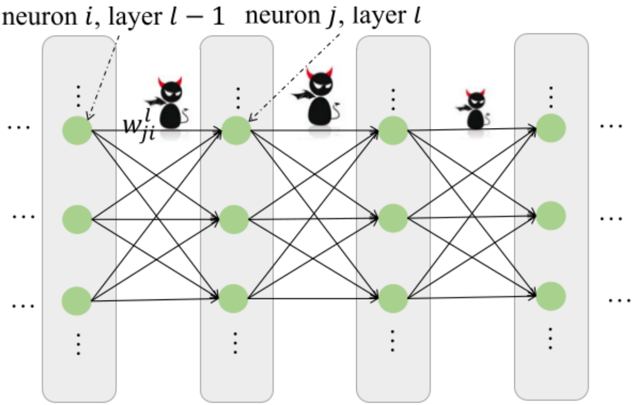


Fig. 6. The error propagates layer by layer and the value changes.

- To easily understand how the error is defined, imagine that the error is represented by a demon icon as depicted in Fig.
- The demon sits at the j-th neuron in the i-th layer. The demon interferes with the neuron's operation when the input to the neuron comes in. It adds a little change (i.e., a noise) to the neuron's weighted input, so that instead of generating only the expected output, the neuron instead outputs the expected output along with the noise.
- Consequently, to minimize the value of the cost function, we need to adjust the values of the weights.

The Values of Weights and Mean Square Error (MSE) for Different Numbers of the Backpropagation Steps

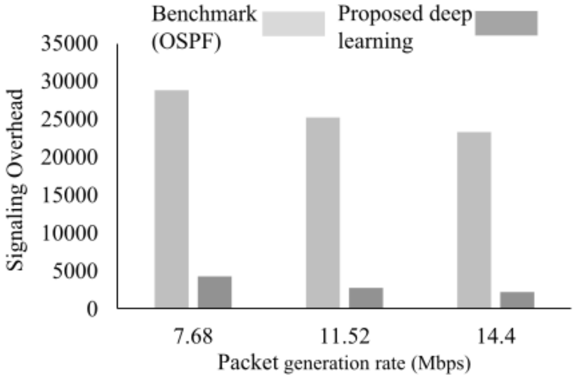
Step	Weight	Value	MSE
100	w_{00}^1	0.825148	2.918418×10^{-5}
	w_{10}^1	-0.600210	
	...	...	
	$w_{11,15}^3$	-0.074882	
	$w_{12,15}^3$	-0.052039	
200	w_{00}^1	0.825954	6.719093×10^{-6}
	w_{10}^1	-0.598363	
	...	...	
	$w_{11,15}^3$	-0.143467	
	$w_{12,15}^3$	-0.121055	
...	...	...	...
500	w_{00}^1	0.825623	9.843218×10^{-7}
	w_{10}^1	-0.597710	
	...	...	
	$w_{11,15}^3$	-0.231773	
	$w_{12,15}^3$	-0.209574	
...	...	...	...
1000	w_{00}^1	0.825115	4.882770×10^{-7}
	w_{10}^1	-0.597787	
	...	...	
	$w_{11,15}^3$	-0.263871	
	$w_{12,15}^3$	-0.241689	

- After the training phase, every router obtains the values of the weights and biases of the DBNs that predict the next nodes for edge routers from itself.

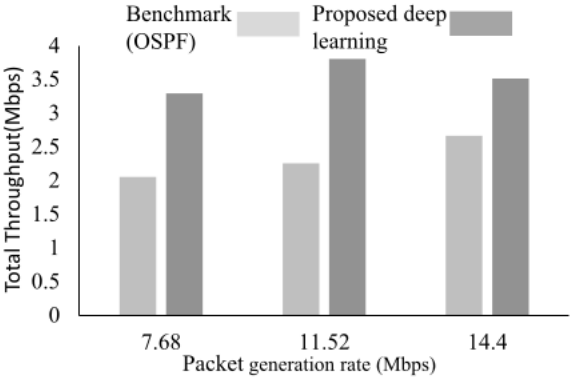
- The next step is that every router forwards these values to the edge routers.

- Since the computations of all the routers were outsourced to a single machine, the evaluation was restricted to a medium scale wireless mesh backbone network comprising 16 routers shown in Fig. rather than a full-scale backbone network topology.Note that this scale of simulation is sufficient as long as it demonstrates that the proposed deep learning system outperforms the conventional routing strategies such as OSPF. The data and control packets sizes are both set to 1Kb. The link bandwidths are set to 8Mbps which is reasonable for this scale of wireless mesh backbone. Every node is assumed to have an unlimited buffer.

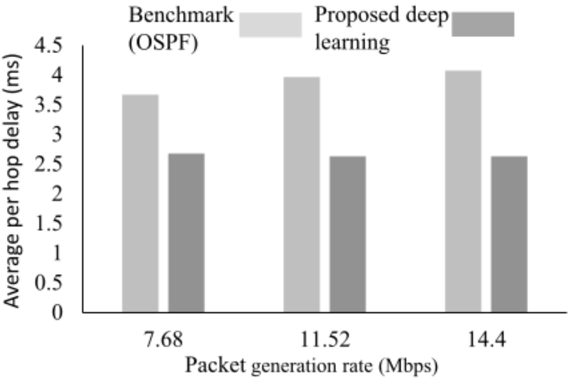
- The overall data packet generating rate in the considered network is varied between 7.68Mbps to 14.4Mbps. For comparison of the adopted deep learning system, OSPF is used as the benchmark method. In the conducted simulations, the time-interval of each path updating phase is set to 0.25s during which signaling is exchanged once.



(a) Comparison of signaling overhead.



(b) Comparison of aggregate throughput.



(c) Comparison of average per-hop delay.

Conclusion

- Deep learning is a new breed of Machine Intelligence technique, which is gaining much popularity and wide use in various computer science fields, such as object recognition,speech recognition, signal processing, robotics, AI gaming, and so forth. However, the application of deep learning in network systems just started to receive research attention.

Thank you