Applied GIS

ISSN: 1832-5505
Vol-6 Issue-02 April 2018

A Review of Deep Learning Techniques for Mobile Applications

Ganesh Aher1, Reena Shrikant Sahane2, Monika Sharad Deshmukh3
Department of Computing, SPPU Pushpalata

Abstract— The need for mobile gadgets, which permeate every facet of modern life, has increased dramatically in recent years. The latest method of development is machine learning, which uses a teach-and-train technique to create a variety of mobile apps. One of the most significant ideas in machine learning is deep learning, which has been shown effective in a variety of real-world contexts. The goal is to use deep learning to create more efficient and user-friendly mobile apps by training them on data collected from users' smartphones. According to the instruction given, the survey data obtained from mobile devices may be used to infer the user's mood disorders. This is how the deep neural network becomes a novel and practical approach to creating mobile apps.

Keywords— Synonyms: Deep Neural Network; Mobile Cloud; Deep Learning; Federated Learning

INTRODUCTION

This There has been a meteoric rise in the use of mobile devices over the last several years, and this trend is expected to continue into the foreseeable future. Mobile devices are expected to reach 5.6 billion, accounting for 21% of all networked devices in 2020 [1]. By the end of 2023, more than 90% of individuals in industrialized nations will possess at least one mobile device. Today's mobile gadgets are changing not just how we go about our everyday business, but also how we live and relate to one another. Machine learning (ML) is also widely utilized in mobile apps for tasks including object identification, language translation, health monitoring, and virus detection. Mobile devices capture a wealth of data on users' behavior, preferences, and habits owing to the devices' frequent interactions with users throughout the day. This data is then used as a resource by machine learning applications. The penetration of mobile apps with ML features among adult users in industrialized nations approaches 60%, according to a survey by Delloitte. Predicted future mobile devices will include machine learning extensively. Recently, deep learning (DL) has been at the forefront of machine learning advancements. DL's exceptional performance has smashed many previous records set by standard machine learning algorithms. Data processing, modeling, and interpretation have all been significantly improved by deep learning. Inspired by deep learning's impressive results, many are trying to use it on mobile devices to provide smarter services. Deep learning is expected to play a crucial role in the future of mobile app creation. Despite the promising outlook, the present investigation into combining deep learning with mobile devices is just at the start. Training and inference are the two main components of deep learning-based applications. Given a DNN model trained for a specific application, we can apply it to the inference task, such as recognizing an image it has never seen, by using the training stage to automatically adjust the trainable parameters in the DNN using the gradient descent algorithms. For mobile devices, none of these jobs is trivial due to their limited processing power and battery life. Training a deep neural network with hundreds of millions of parameters may quickly introduce huge resource demands that are well above the capability of mobile devices. Instead of training a DNN model on mobile devices, the present study focuses on how to make maximum use of the scattered data created by users' mobile devices while maintaining their security, secrecy, and privacy. The inference phase when using a DNN model on a mobile device may be just as tough as the training phase when attempting deep learning on a mobile device. The huge DNN models exceed the insufficient on-chip memory of mobile devices, necessitating the use of off-chip memory—a solution that comes with the downside of much higher energy consumption. One significant negative is that the huge dot products significantly increase the workload of processing units. When a deep learning program is running, it may quickly take control of the system's energy consumption[15]. In order to overcome these obstacles, researchers in both academia and business have investigated how to properly apply DNN on mobile devices. This research is crucial for advancing deep learning on mobile devices. In this article, we will discuss the progress that has been made in implementing deep learning on mobile devices, as well as the obstacles that still need to be overcome. How to efficiently deploy DNNs on mobile devices and make use of the data supplied by individuals' mobile devices are the key topics of debate. In addition, we will launch cutting-edge projects pertaining to deep learning on mobile devices, with a focus on the practical uses of the data provided by these devices.

LITERATURE REVIEW

In the middle of the twentieth century, Artificial Neural Network was used to present the first biologically inspired models of shallow ANN [8].Despite their inability to pick up new skills, these networks were responsible for introducing the first algorithms for supervised training [9].[10][11]

The ANN were popular between ’70 and ’80, beginning with the advent of back-propagation learning algorithm and Rumelheart al[86] has done work from which generating usable representation of input data in hidden layers of NN. As a result of the BP-trained ANN's inability to provide a complete answer to any machine learning challenge, the enthusiasm of the research community waned. For pre-training a network using an auto encoder and unsupervised learning, Ballard introduced hierarchies in 1987. [1]. LeCun's 1989

ISSN: 1832-5505
Vol-6 Issue-02 April 2018

enforcement of the BP NN to convolution is a crucial component of today's deep learners' use of NNs with adaptive connections[7].

LeCun's network architecture exhibits topographic structure due to the presence of two kinds of layers, subsampling and convolution. Fundamental A few years later, the Hochreiter posed the first deep learning challenge. The work of Benigoetal et al. [6] against a solution for deep network learning was prompted by this issue, commonly known as the long time lag problem. [2][6]. In the same year, the Cresceptron model established the widely-accepted notion of Max-pooling (MP) layers in neural architecture for contemporary DL. In order to garner interest from the ML community, the non-neural SVM [12] technique was used. Unsupervised multi-stage auto-encoders, which simulate the pre-training stage before fine-tuning, have attracted a lot of interest in the same time period.[3][4] Reduce training time by taking use of efficient parallel algorithms made possible by compellingly priced high-performance graphics processing units (GPUs) in the same year. Due to the rapid development of efficient algorithms, sophisticated network models, and ever-increasing computational models, DL technologies have spread rapidly during the last decade. In 2012, Alex Krizhevsky et al., popularly known as AlexNet, combined MPCNNs with GPU and achieved strong results in the ImageNet benchmarking project. AlexNet's five convolution layers were used in the network's construction. Max-pooling layers, dropout layers, and 3 fully-connected layers were used to classify 1000 categories, and in the years that followed, more advanced designs were developed.ZF Net [14], suggested by M. Zeiler and R. Fergus in 2013, is an attempt to extract the inner mechanism of a network by predicting the feature maps. Inception architecture, often known as GoogleNet, was the next major development in the design of flexible client networks.[13]. With the help of Inception, people started to realize that CNN layers didn't have to be stacked in a certain order. ResNet, a Deep Network with 152 layers, was released in the same year it won the 2015 ILSVRS competition. With a maximum 3.6% margin of error. The notion of generative adversarial networks, initially described by Ian Good enough et al in [5], is especially noteworthy, since it represents one of the most exciting findings of the last several years.In this approach, two networks compete in a game where the generator seeks to learn the distribution of the input data in order to trick the classifier (the discriminator), while the la.er tries to prevent the generator from succeeding. At convergence, the generator's samples can no longer be distinguished from the training samples.

SYSTEM OVERVIEW:

There are various challenges to feed mobile devices with deep learning approach. To this there are three aspects training, inferringand applications.

1. Distributed Data Training

In this training process the data is collected from the mobile devices to create a smart mobile application. To work on this data it is required to use DNN which uses cloud data centres and high performance computers. While doing this the restriction arises todirectly use individual's data as there can be privacy issues. The solution is to propose the distributed training scheme such that the data is taken without exposing them.The main process of training is to fit the training parameters in DNN. In proposed algorithm distributed selective SGD, whichever participants are having local dataset they participates in the DNN training. From this each of the participants local DNN model is trained by using every individual local dataset by standard SGD. During local training there is no communication with other participants and global parameter server. Every time the global parameters are updated by using gradients of selected parameters which are uploaded to global parameters. The global parameter server is used by participants to download the most recent parameters to update their local DNN model. This is how the distributed training for DNN works without explicitly sharing of participants' local data to collect data from mobile devices.

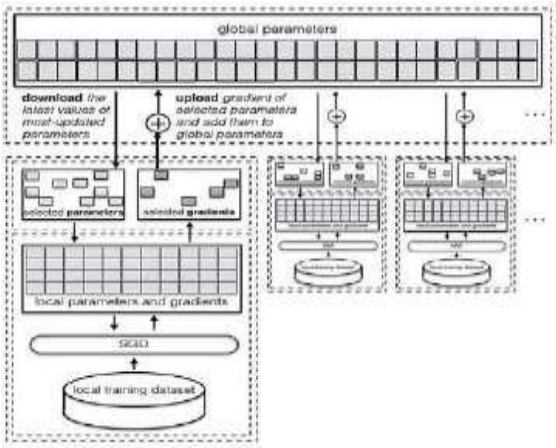

Fig 1: Framework Of Distributed Selective SGD

2. Federated Training on Mobile Devices

In federated training each mobile device can participate and download the shared DNN model from cloud and improve the learning process of data from local data and after learning process is over it can then reflect the changes to the cloud to improve the shared model. This training process tries to fully utilize the mobile device processors to generate better quality updates

Federated training achieves higher efficiency than the naively distributed SGD. For this training to work

ISSN: 1832-5505
Vol-6 Issue-02 April 2018

successfully it is required that mobile device should be idle, plugged etc to avoid problem with the performance.

3. Privacy-Preserving Training

During the distributed training process there are chances for the local training data to exploit any of the individual information so to overcome this privacy measures should be applied. Differential privacy-preserving is the which is used for data analysis, which gives assurance of preserving sensitive data of the participant of training process. Various work has been proposed to take care of this privacy factors but somewhere it lacks the proper differential privacy, so to overcome this a non-private federated training aremade which are as follows:

a. The participants are selected independently with probability p instead of always selecting the fixed participants.
b. L2 norm is applied for the updates generated by the participants.
c. To apply moment's accountant an estimator is used for weighted aggregation.
d. The final average update is calculated by adding sufficient Gaussian noise.

This is how by applying these modifications the training process becomes good approach for DNN model

CONCLUSIONS

The use of deep learning on mobile devices is a hot area in research and development. Expect more development of deep learning based mobile apps in the next years as demand for intelligent services on mobile devices rises. The study of deep learning on mobile devices, however, is only getting started. It is not a simple task to allow deep learning services on mobile devices due to the inherent tension between the high resource need of DNN and the restricted capability of mobile devices. In this study, we examine three key features of the most influential publications on mobile deep learning: (1) mobile data training, (2) mobile inference, and (3) mobile deep learning applications. We identify the key issues and the best ways to address them. The report also provides a brief summary of recent research.

REFERENCES

[1] Dana H. Ballard. Modular learning in neural networks. In K. Forbus and H. Shrobe, editors, Proceedings of the Sixth National Conference on Arti€cial Intelligence, pages 279–284. San Francisco, CA: Morgan Kaufmann.

[2] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is di ▮cult. Trans. Neur.Netw., 5(2):157–166, 1994.

[3] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy layer-wise training of deep networks.In P. B. Sch¨olkopf, J. C. Pla., and T. Ho.man, editors, Advances in Neural Information Processing Systems 19, pages 153–160. MIT Press, 2007.

[4] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? J. Mach. Learn. Res., 11:625–660, 2010.

[5] an Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, Advances in Neural Information Processing Systems 27, pages 2672–2680. Curran Associates, Inc., 2014

[6] S. Hochreiter. Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut f¨ur Informatik, Lehrstuhl Prof. Brauer, Technische Universit¨at M¨unchen, 1991.

[7] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwri.en zip code recognition. Neural Comput., 1(4):541–551, 1989.

[8] Warren S. McCulloch and Walter Pi.s. A logical calculus of the ideas immanent in nervous activity. .e bulletin of mathematical biophysics, 5(4):115–133, 1943.

[9] Kumpati S. Narendra, Senior Member, and M. A. L. .athachar. Learning automata - a survey. IEEE Trans. Systems, Man., Cybernetics, pages 323–334, 1974.

[10] F. Rosenbla.. .e perceptron: A probabilistic model for information storage and organization in the brain. Psychological Review, pages 65–386,1958.

[11] F. Rosenbla.. Principles of neurodynamics: perceptrons and the theory of brain mechanisms. Spartan Books, 1962.

[12] Vladimir N. Vapnik. .e Nature of Statistical Learning .eory. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[13] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Sco. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In Computer Vision and Pa.ern Recognition (CVPR), 2015.

[14] Ma.hew D. Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks, pages 818–833. Springer International Publishing,2014.

[15] Mrs. S. A. Bhavsar Mr. Dattatray S. Shingate "Securing Mobile System Locations by Anonymous Server Design based on k Optimal Principle", international journal of pure and applied research in engineering and technology Volume 02,Issue 08 PP 208-218