INTRODUCTION TO DEEP LEARNING USING THE INTEL® NEON™ FRAMEWORK

DESCRIPTION

The course examines the construction and the performance analysis of deep neural networks using the Intel® neon™ Framework.

The following topics are covered:

1. Introduction to deep learning.
2. Multilayered fully-connected neural networks.
3. Introduction to the Intel® neon™ Framework.
4. Convolutional neural networks. Deep residual networks.
5. Transfer learning of deep neural networks.
6. Unsupervised learning: autoencoders, deconvolutional networks.
7. Recurrent neural networks.
8. Introduction to the Intel® nGraph™.

The course is practice oriented. There are 8 lectures (1.5 hours each) and 5 individual consultations in groups of 2-3 people (for each group). Lectures are held in plain lecture or master class (tutorial) form. The presentation of the theoretical material in most lectures/master classes is supported by examples of developing a deep neural network architecture using the Intel® neon™ Framework. The problem for which deep models are constructed is comprehensive and covers the entire lecture part, with the exception of an introductory lecture of a survey nature. The practice of the course is structured as follows: students are divided into groups of 2-3 people. Each group chooses a separate problem and tries to achieve the maximum quality by constructing different types of deep architectures and modifying their internal structure. Students follow the provided tutorials that represent step-by-step deep model development using the Intel® neon™ Framework. The mode of collective development is simulated. The final control of knowledge assumes presentation of the developed project with demonstration of quality/performance measurements of proposed deep neural networks.

The course is aimed at engineers, teachers and researchers, as well as postgraduate students and students of higher educational institutions.

PRELIMINARY REQUIREMENTS

The course is aimed at students who have basic programming skills in the scripting programming language Python. Along with this, the course requires theoretical knowledge in the field of optimization methods, probability theory, image processing and computer vision.

THE COURSE STRUCTURE

The course schedule is as follows.

Topic	**Content**	**Hours**	
		Lecture	**Practice**
Introduction to deep learning (LECTURE 1)	The notion of deep learning. Biological fundamentals of deep learning. Examples of practical problems.	**2**	

	Classification of deep models.		
Multilayered fully-connected neural networks (LECTURE 2)	The structure of fully-connected neural networks (FCNN), types of activation functions. Training problem of FCNN, loss function. Backpropagation method.	**2**	
Preprocessing and converting data to HDF5 format for the Intel® neon™ Framework (PRACTICAL CLASS 0)	Preliminary practice to prepare dataset for the subsequent practical classes.		**4**
Introduction to the Intel® neon™ Framework (LECTURE 3)	Introduction to the Intel® neon™ Framework. Installation. The structure of application for training/testing of the single-layer fully-connected neural network using the Intel® neon™ Framework.	**2**	
The development of fully-connected neural networks using the Intel® neon™ Framework (PRACTICAL CLASS 1)	Problem statement for the laboratory works. Development of architectures of fully-connected networks with a different number of hidden layers and the number of hidden elements on each layer. Developing scripts for training/testing the proposed architectures. Carrying out experiments, collecting performance results.		**4**
Convolutional neural networks. Deep residual networks (LECTURE 4)	The structure of convolutional layer and network. Example of training/testing a single-layer convolutional network using the Intel® neon™ Framework. Deep residual networks, a typical structural block, an example of a residual network.	**2**	
The development of convolutional neural networks using	Development of convolutional network architectures with different number of hidden layers and filter parameters on each		**4**

the Intel® neon™ Framework (PRACTICAL CLASS 2)	layer. Developing scripts for training/testing the proposed architectures. Carrying out experiments, collecting performance results.		
Transfer learning of deep neural networks (LECTURE 5)	Description of the general approach underlying the transfer learning in deep neural networks. An example of transfer learning application using the Intel® neon™ Framework.	**2**	
Application of transfer learning to solve a given problem using the Intel® neon™ Framework (PRACTICAL CLASS 3)	Selection of the original problem (connected with a given problem) and a corresponded trained model. Modification of the network architecture for a given problem. Complete learning of the parameters of all network layers with arbitrary initialization. Training of all layers of parameters of all layers of the network with initialization, obtained as a result of learning the model to solve the original problem. Learning only the last layers (modified) of the network with initial initialization, obtained as a result of learning the model to solve the original problem.		**4**
Unsupervised learning: autoencoders, deconvolutional networks (LECTURE 6)	Unsupervised learning methods. The concept of an autoencoder, a stack of autoencoders, deconvolutional networks.	**2**	
Initial pre-training the weights of the most perspective architectures of fully-connected networks for the subsequent solution of a given problem in supervised style using the Intel® neon™ Framework (PRACTICAL CLASS 4)	Selection of several fully-connected neural networks. Developing a stack of autoencoders. Training of the developed architectures. Application of the obtained initial weights for training the network in supervised style to solve a given problem.		**4**

Recurrent neural networks (LECTURE 7)	The general structure of the model. Deploying a recurring network in time. Recurrent networks training. Long short-term memory network. An example of training/testing a simple recurrent network using the Intel® neon™ Framework.	**2**	
The development of recurrent neural networks using the Intel® neon™ Framework (PRACTICAL CLASS 5)	Development of architectures of recurrent neural networks with a different number of hidden layers and the number of hidden elements on each layer. Developing scripts for training/testing the proposed architectures. Carrying out experiments, collecting performance results.		**4**
Efficient execution of neural networks. The Intel® nGraph™ overview (LECTURE 8)	Introduction to the Intel® nGraph™. The neon™ frontend to Intel® nGraph™	**2**	
EXAM: project presentation			
	Total:	**16**	**24**

COURSE MATERIALS

Course materials:

1. *Syllabus*.
2. *Lectures*: 8 lectures. There are a presentation and an annotation for each lecture (2-3 pages with the list of recommended literature).
3. *Practical classes:* instructions for implementing practical tasks; a case study in source code that demonstrates the implementation of practical tasks, and it is supported by a step-by-step description; list of possible practical assignments.

   We plan to use the IMDB-WIKI dataset [https://data.vision.ee.ethz.ch/cvl/rrothe/imdb-wiki] to represent the implementation of the practical tasks #1-5 on the problem of gender prediction. Some practical tasks will require a step of image preprocessing and converting markup to the Intel® neon™ Framework format.

LANGUAGE

Russian, English

LICENSE

AUTHORS

Kustikova Valentina Dmitrievna, Phd, lecturer, department of Computer software and supercomputer technologies, Institute of Information Technologies, Mathematics and Mechanics, Nizhny Novgorod State University. Lead and developer.

Zolotykh Nikolai Yurievich, Dr., Prof., department of Algebra, geometry and discrete mathematics, Institute of Information Technologies, Mathematics and Mechanics, Nizhny Novgorod State University. Scientific adviser.

Zhiltsov Maxim Sergeevich, master of the 1st year training, Institute of Information Technology, Mathematics and Mechanics, Nizhny Novgorod State University. Developer.