Convolutional Neural Networks for Real-Time Data Classification

Narek, Abroyan
National Polytechnic University of Armenia
Yerevan, Armenia
e-mail: narek.abroyan@gmail.com

Robert, Hakobyan
National Polytechnic University of Armenia
Yerevan, Armenia
e-mail: rob.hakobyan@gmail.com

ABSTRACT

In this paper, we provide a method of classification of real-time data by using convolutional neural networks. During the last several years, artificial intelligence researchers and specialists achieved notable results in visual processing tasks by using new methods and approaches of deep learning, such as convolutional neural networks. The goal of this work is experimenting and providing a new approach of using convolutional neural networks for classification of non-stationary data.

Keywords

Deep learning, convolutional neural networks, real time data, classification.

1. INTRODUCTION

Since 2006, deep machine learning (more commonly called deep learning), has appeared as a new field of machine learning research [1]. During the past several years, the techniques and methods that have been developed from deep learning research have revolutionized and started a new era of machine learning and artificial intelligence. There are three important reasons which made deep learning so popular today - the chip processing abilities have been drastically increased (e.g., general-purpose graphical processing units or GPGPUs), the size of data for training have been significantly increased, and some new discoveries were made in the field of machine learning and signal/information processing research [1]. The usage of convolutional, recurrent neural networks, restricted Boltzmann machines, and deep belief networks hugely improved the results of machine and deep learning problems. However, most researches that took place now are mostly devoted to visual, natural language, signal processing. Thus, researchers do not often talk about real-time machine learning, which has some specific properties [2]. However, the usage of real-time systems becomes more and more widespread. Thus, it is quite effective and convenient to use machine learning in real time systems for elaborating a huge amount of newly generated data. In this work, we continue the research started in [3]. As it was turned out in [3], just by switching from classical feed forward to convolutional neural networks mainly keeping the structure of neural network, the performance will not be improved. Hence, there is a need to take extra steps and pick a correct architecture for the neural network. Evenmore, as no free lunch theorem states, no algorithm performs universally better than any other [4]. For that reason the architecture should be picked up individually for each specific problem. In this paper we are concentrating on convolutional neural networks and propose a new method for classification of real-time non-stationary data.

2. REAL-TIME DATA

In this work by saying real-time system, we mean that its total correctness depends not only upon its logical correctness, but also upon the time in which it is performed [2]. A lot of examples of such data are in financial sphere, as an example of such data could be stock prices that rapidly change over time. Another example of real-time system can be financial transaction fraud detection system. That is, all orders in stock exchange market are processed and tested for not being fraud. One way of doing that, is using a Gaussian distribution [5]. After receiving an alert about a possible fraud operation, a qualified person must decide whether that action is normal or not. In fact, that person solves a real-time classification problem, which can be modelled into a machine learning problem. Another and very popular example is deciding of credit approval, i.e., whether a person should be allowed to take a credit from a bank or not. In all above cases, correctness of data and approval decision strongly depend on time. Data of a market order or a bank client sometimes can qualitatively be changed later and may not be relevant to the training data. For now, we are going to train our model based on historical data and do classification in real time. For simplicity we will not take into account the mentioned fact that historical data may become irrelevant over some time (this is going to be a topic for further research). The basic structure of the described system is presented in Fig. 1.

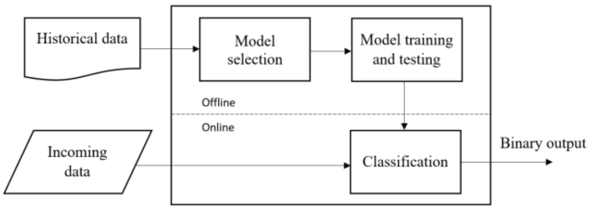

Fig. 1 The illustration real time classification system

In this work, experiments of neural networks are mainly performed for examples of the credit approval and fraud detection problems.

3. METHODOLOGY

3.1. Convolutional Neural Networks and Feature Selection

Convolutional neural networks are a specific type of neural networks that are best known for processing data with grid-like structure [4]. However, they can be used also for one-dimensional data such as time-series (i.e., one-dimensional grid, the samples of which are at regular time intervals). In case of images, data can be represented as a two-dimensional grid of pixels [4]. The name "convolutional neural network" is used, because the network uses a mathematical operation of convolution [4]. Convolution is a specialized kind of linear operation. Convolutional networks are simply neural networks that use convolution operation instead of general matrix multiplication in at least one of their layers [4].

In addition, convolutional networks recognize two-dimensional shapes with a high degree of invariance to translation, scaling, and other forms of distortion [6]. That task is performed as a supervised problem with the help of a specific structure, which has some limitations, includes the following forms of restrictions [6]

- Feature extraction. Each neuron takes its inputs from a local receptive field in the previous layer, thus forcing it to extract local features based on that local field. When a feature has been extracted, its exact location becomes not important [6].
- Feature mapping. Each computational layer of the network consists of several feature maps. Each feature map has the same dimensionality as the input data and each neuron within the single feature map uses the same shared weights for input [6].
- Subsampling/pooling. After each convolution layer, usually (but not mandatorily) there is a special computational layer, which performs local averaging or maximizing subsampling/pooling operation. As a result of that operation, the resolution of the feature map is reduced. Also, due to this operation the sensitivity of the feature map's output is reduced to some forms of distortion [6].

Eventually, the convolutional neural network learns to extract its own features automatically [6] and more abstract features are learned from the lower level ones [2]. Typically, the most computationally expensive part of convolutional network training is learning the features. The fully connected network with output layer is usually relatively inexpensive, because after passing through several layers of pooling/subsampling, already the small number of features provided as input to the fully connected network.

Some typical examples of a convolutional networks for image processing are represented in [6] for handwritten characters recognition and an implementation in [7] for handwritten digits recognition.

After all this we can assume that convolutional neural networks are a good option for the classification of credit card data. They are fast, use less parameters, extract features themselves. Wherein, the last property is quite important, because it can free us from feature hand crafting.

There is already some research in this area [8], where credit card fraud detection is done by using convolutional neural networks. It is stated that experimental results from the real transaction data of a commercial bank show that the proposed method performs better than other state-of-art methods [8]. However, the input data for each training example were reshaped from one dimensionality to two for making possible of using two-dimensional convolutional neural networks. Besides, quite big neural network was used, which requires a lot for resource for training. In scope of this work we try to reach better results with smaller convolutional neural networks.

In this work one-dimensional convolutional neural networks are used. On the one hand, using one dimensional neural networks we will gain more performance and use less parameters. On the other hand, as each example is just a one-dimensional vector of double numbers, it is possible to use one-dimensional convolutional neural network without reshaping input data. The high-level structure of the used neural network is presented in Fig. 2.

The explanation of enumartions in Fig. 2 is given below

1. A single training sample with all features
2. Convolutional operation
3. Max or average pooling/subsampling operation
4. Visualization of the fact, that more than one convolution-pooling layers can be present in a single neural network
5. Fully connected network

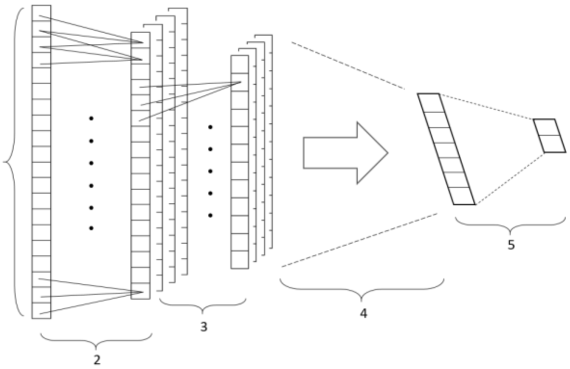

Fig. 2 High level structure of one-dimensional convolutional neural network

So, as it is shown in the Fig. 2, a single training sample is being used for every training iteration. Convolution operation is being performed on features of each training sample with multple filters. In a result of this, several feature maps are being formed. For each of those feature maps max or averege pooling operation is performed. In a single neural network more than one convolutional-pooling opeartion can be used. A simple fully connected network follows after all convolution-pooling operations.

This approach gives an advantage of feature automit extract, i.e., if there is a need of classification of some credit card data and there are a lot of possible features, this approach should solve that problem by filtering useful features. Thus, there will be no need to manually choose features.

3.2. Experiments and Results

For experiments and implementation of neural networks, Keras library with TensorFlow backend was used [7]. As an initial experimental data set, "German Credit Data" data set has been used. The whole data set and detailed information can be found in [9]. As data set that is being used, is not big, k-fold cross validation algorithm [4] is being used for choosing hyperparameters. Binary crossentropy [4] was chosen as a loss function. Rectified linier unit activation function was used for hidden layers [4]. As we solve binary classification problem, we use the sigmoid activation function instead of softmax for the output unit [4]. For making the model to generalize well, the dropout regularization method was used [4]. All the mentioned hyperparameters have been picked up by considering the best practices with empiric origins [1, 4, 5]. For comparing our experiments with the already known result in [8], we are going to measure our model performance with

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

score [5]. As it is known, there should be a trade-off between precision and recall. The latter values can be adjusted by changing the decision threshold. In a result of experiments, up to 0.8 F_1 score was achieved just by using one convolution-pooling layers and one hidden layer from the fully connected network. This result is better than the best result in [8].

Moreover, for making sure that our results perform and generalize well on another similar credit card transaction fraud detection data set. Detailed information for this dataset can be found in [10]. Here also up to about 0.8 F_1 score was achieved, with the same neural network architecture.

4. CONCLUSION

In this paper, we proposed a method of classification of non-stationary data classification using one-dimensional convolutional neural networks. In a result of experiments we reached up to 0.8 F_1 score on different data sets with a small neural network, which is better than the latest already known method proposed in [8].

REFERENCES

[1] L. Deng, D. Yu, "Deep Learning: Methods and Applications", *Foundations and Trends in Signal Processing, Vol. 7, Nos. 3–4,* pp. 197–230, 2014.

[2] N. Abroyan, R. Hakobyan, "A Review of the Usage of Machine Learning in Real-time Systems", *Proceedings of NPUA, Information technologies, Electronics, Radio engineering, № 1,* pp. 46–54, 2016

[3] N. Abroyan, "Classification of Real-time Data Using Deep Learning", *Proceedings of the 13th International Conference of Science and Technology, New Information Technologies and Systems, Penza, Russia,* pp. 109-112, 2016

[4] I. Goodfellow, Y. Bengio, A. Courville, "Deep Learning", *Cambridge, Massachusetts, The MIT Press*, 2016

[5] N. Andrew, "CS 229 Machine Learning Course Materials", *Stanford University,* [http://cs229.stanford.edu/materials.html], 2016

[6] S. Haykin. "Neural Networks and Learning Machines", *3rd ed., McMaster University, Hamilton, Ontario, Canada,* 2009.

[7] F. Chollet, "Keras: Deep Learning library for Theano and TensorFlow", *Github*, [https://github.com/fchollet/keras], 2015

[8] K. Fu, D. Cheng, Y. Tu, L. Zhang, "Credit Card Fraud Detection Using Convolutional Neural Networks", *Proceedings of 23rd International Conference, ICONIP, Part III, Kyoto, Japan,* pp. 483-490, 2016

[9] M. Lichman, "UCI Machine Learning Repository" [http://archive.ics.uci.edu/ml], *Irvine, CA: University of California, School of Information and Computer Science,* 2013

[10] A. D. Pozzolo, O. Caelen, R. A. Johnson, G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification", *In Symposium on Computational Intelligence and Data Mining (CIDM), IEEE*, 2015