Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 00 (2017) 000–000

Procedia
Computer Science

www.elsevier.com/locate/procedia

Information Technology and Quantitative Management (ITQM 2016)

Hybrid learning net: a novel architecture for fast learning

Ying Liu[a], Chao Xiang[a]

[a]*University of Chinese Academy of Sciences, Beijing 100049, China*

Abstract

Currently, neural networks have succeeded in object recognition tasks based on images, natural language translation, and voice recognition, to name a few. However, neural nets are customly built for different applications and vary a lot in achitectures and model hyperparameters like learning rate and parameter initialization. To make it worse, these hyperparameter settings generally play a big role in performances of training and testing, and the best settings for specific applications are so far only available by manually repeatly trying different configurations which is really huge work. We, thereby, present a novel neural network achitecture, called Hybrid Learning Net(HLN), with Self Organizing Maps(SOMs) aided to learn from samples in both unsupervised and supervised way, targeting to achieve a much faster net learning for general applications with good robustness to a few key hyperparameters such as the parameter initialization and the net strcuture variation. We've also experimented our architecture over the MNIST dataset, it has proved the impressive promotion on both training and testing phases of general applications, and we have unexpectedly discovered some interesting facts about neural network trainings, such as neuron activation sparsity is somehow strongly related to the training loss within certain cases.

Selection and/or peer-review under responsibility of ITQM2016.

Keywords: hybrid learning; neural network; general application; sparsity; SOM; MNIST; fast learning

1. Introduction

Since the first mathematical model of artificial neural network was proposed in 1943[1], the neural network has been designed into many architectures, such as Convolutional Neural Networks(CNNs) for image recognition [2], Recurrent Convolutional Neural Networks(R-CNNs) for object detection in videos[3], and Long Short Term Memorys(LSTMs) for speech recognition [4] and many, many more to make it a list. These specially designed neural network models are trained by lots of efficient methods with tons of carefully chosen little skills which we may call tricks. Such neural network architectures suit well for their specific applications but may have plain or worse performances on others, and their best performances rely heavily on hyperparameter configuration in general. Thus, it is an in-demand job to propose a relatively universal architecutre that enables equal or similar performances among varied applications.

We notice that, although there're plenty of choices to train a neural network, literally all these methods can be reduced into 3 categories, supervised learning [5], unsupervised learning [6], and the semi-supervised [7]. In supervised learning, one can only train a model from labeled samples, however in real applications, labeling a

*Corresponding author. Tel.: +86-183-0114-2368;
E-mail address: xiangchao215@mails.ucas.ac.cn.

large dataset is a tough and costly task, the unlabeled ones are therefore the primary data available. To make use of the majority unlabeled data, a few unsupervised training algorithms arose. These unsupervised learning methods, denoted as pretraining, attempt to produce an optimized parameter initialization [8]. Thus such unsupervised techniques can only be applied before the supervised training phase, once the supervised begins, the unsupversied learning will become unavailable for the model. While the semi-supervised learning allows the model to learn from both labeled and unlabeled samples at the same time, but the technique is way too complicated, and not neural-network-integreted [9].

Here we introduce a new architecture that learn in both supervised and unsupervised ways at the same time, and requires no extra techniques on training. Our architecture, Hybrid Learning Nets(HLNs), made of decoupled hidden layer consisting a Self Organizing Map(SOM) and a fully connected layer, training in the simplest way of Backprop, demonstrate much faster learning capability and remain robust to parameter initialization and network configuration such as the net depth of layer.

1.1. Semi-supervised learning for neural network

A key assumption in semi-supervised algorithms developed so far, is the structure assumption: two samples with similar distribution on the same mapping structure tend to have high probability of belonging to the same class. Based on this assumption, one can use large unlabeled data to uncover such structures. There're already a few algorithms dedicated to do this, such as cluster kernels [10], Low Density Separation(LDS) [11], label propagation [12], to name a few. In such algorithms, designing a regularizer to enable the model to learn the representation or structure of raw data, in order to improve the supervised learning performance, becomes the key point.

To get a clear idea of how such semi-supervised learning algorithms work, we will give a brief recall to some. Let's firstly focus on the general algorithm description of semi-supervised learning. Given a set of unlabeled samples, $S = \{x_1, \cdots, x_N\}$, and the similarity labels between any x_i and x_j, $\{W_{ij} | i, j = 1, \cdots, N\}$, we're to find the best embedding function, $f(x)$, for each sample x_i, to minimize:

$$\sum_{i=1}^{N}\sum_{j=1}^{N} L(f(x_i), f(x_j), W_{ij}) \tag{1}$$

to explain it,

- $L(\cdot)$ is the loss function of 3 variables: $\left\langle f(x_i), f(x_j), W_{ij} \right\rangle$, such as

$$L(f(x_i), f(x_j), W_{ij}) = \max(0, \|f(x_i) - f(x_j)\| - W_{ij})$$

- $f(x)$ is the embedding function, it tries to produce a vector from x_i, similar to that of x_j with $W_{ij} = 0$, and disimilar with $W_{ij} = 1$.
- $W_{ij} \in \mathbf{R}^d$ is the similarity label matrix on $x_1, x_2, \cdots, x_N$

1.2. Tables

All tables should be numbered with Arabic numerals. Headings should be placed above tables, left justified. Leave one line space between the heading and the table. Only horizontal lines should be used within a table, to distinguish the column headings from the body of the table, and immediately above and below the table. Tables must be embedded into the text and not supplied separately. Below is an example which authors may find useful.

Table 1. An example of a table.

An example of a column heading	Column A (t)	Column B (T)
And an entry	1	2
And another entry	3	4
And another entry	5	6

1.3. Construction of references

References should be added at the end of the paper, and its corresponding citation will be added in the order of their appearance in the text. Authors should ensure that every reference in the text appears in the list of references and vice versa. Indicate references by [1], [2-3] in the text. The actual authors can be referred to, but the reference citation(s) must always be given. Some examples of how your references should be listed are given at the end of this template in the 'References' section, which will allow you to assemble your reference list according to the correct format and font size.

1.4. Section headings

Section headings should be left justified, with the first letter capitalized and numbered consecutively, starting with the Introduction. Sub-section headings should be in capital and lower-case italic letters, numbered 1.1, 1.2, etc, and left justified, with second and subsequent lines indented. You may need to insert a page break to keep a heading with its text.

1.5. General guidelines for the preparation of your text

Avoid hyphenation at the end of a line. Symbols denoting vectors and matrices should be indicated in bold type. Scalar variable names should normally be expressed using italics. Weights and measures should be expressed in SI units. Please title your files in this order conferenceacrynom_authorslastname.pdf

2. SOMs-embedding for deep architecture

A recall to SOMs.
How we construct deep architecture with SOMs-embeddings.

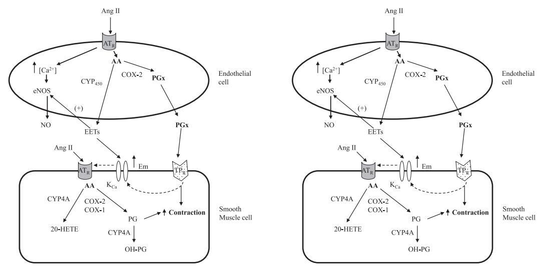

Fig. 1. (a) first picture; (b) second picture.

The figure number and caption should be typed below the illustration in 8 pt and left justified. For more guidelines and information to help you submit high quality artwork please visit: http://www.elsevier.comwps/find/authorsview.authors/authorartworkinstructions. Artwork has no text along the side of it in the main body of the text. However, if two images fit next to each other, these may be placed next to each other to save space, see Fig 1. They must be numbered consecutively, all figures, and all tables respectively.

2.1. Footnotes

Footnotes should be avoided if possible. Necessary footnotes should be denoted in the text by consecutive superscript letters. The footnotes should be typed single spaced, and in smaller type size (8 pt), at the foot of the page in which they are mentioned, and separated from the main text by a short line extending at the foot of the column. The 'Els-footnote' style is available in this template for the text of the footnote.

Equations and formulae should be typed and numbered consecutively with Arabic numerals in parentheses on the right hand side of the page (if referred to explicitly in the text),

$$\begin{aligned} X_r &= \dot{Q}^{"}_{rad} \Big/ \left(\dot{Q}^{"}_{rad} + \dot{Q}^{"}_{conv}\right) \\ \rho &= \frac{\vec{E}}{J_c(T = \text{const.}) \cdot \left(P \cdot \left(\dfrac{\vec{E}}{E_c}\right)^m + (1 - P)\right)} \end{aligned} \tag{2}$$

They should also be separated from the surrounding text by one space.

3. Online licence

All authors must Transfer the Online licence before the article can be published. This transfer agreement enables Elsevier to protect the copyrighted material for the authors, but does not relinquish the authors' proprietary rights. The copyright transfer covers the exclusive rights to reproduce and distribute the article, including reprints, photographic reproductions, microfilm or any other reproductions of similar nature and translations. Authors are responsible for obtaining from the copyright holder permission to reproduce any figures for which copyright exists.

The citation must be used in following style: [13], [14], [15], [16] and [17].

Acknowledgements

These and the Reference headings are in bold but have no numbers. Text below continues as normal.

References

[1] W. S. McCulloch, W. Pitts, A logical calculus of the ideas immanent in nervous activity, Bulletin of mathematical biology 5 (4) (1943) 115–133.
[2] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural information processing systems, 2012, pp. 1097–1105.
[3] R. Girshick, Fast r-cnn, in: Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
[4] A. Graves, N. Jaitly, A.-r. Mohamed, Hybrid speech recognition with deep bidirectional lstm, in: Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on, IEEE, 2013, pp. 273–278.
[5] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, L. D. Jackel, Handwritten digit recognition with a back-propagation network, in: Advances in neural information processing systems, 1990, pp. 396–404.
[6] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, Journal of Machine Learning Research 11 (Dec) (2010) 3371–3408.
[7] O. Chapelle, B. Scholkopf, A. Zien, Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews], IEEE Transactions on Neural Networks 20 (3) (2009) 542–542.
[8] Q. V. Le, Building high-level features using large scale unsupervised learning, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, 2013, pp. 8595–8598.
[9] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, C. D. Manning, Semi-supervised recursive autoencoders for predicting sentiment distributions, in: Proceedings of the conference on empirical methods in natural language processing, Association for Computational Linguistics, 2011, pp. 151–161.
[10] O. Chapelle, J. Weston, B. Schölkopf, Cluster kernels for semi-supervised learning, in: Advances in neural information processing systems, 2003, pp. 601–608.
[11] O. Chapelle, A. Zien, Semi-supervised classification by low density separation., in: AISTATS, 2005, pp. 57–64.
[12] X. Zhu, Z. Ghahramani, Learning from labeled and unlabeled data with label propagation.
[13] L. A. Aamport, The gnats and gnus document preparation system, G-Animal's Journal.
[14] L. A. Aamport, The gnats and gnus document preparation system, G-Animal's Journal 41 (7) (1986) 73, this is a full ARTICLE entry.
[15] L. A. Aamport, The gnats and gnus document preparation system, in: G-Animal's Journal [16], pp. 73+ (1986) 73+, this is a cross-referencing ARTICLE entry.
[16] G-Animal's Journal 41 (7), the entire issue is devoted to gnats and gnus (this entry is a cross-referenced ARTICLE (journal)).
[17] D. E. Knuth, Fundamental Algorithms, Addison-Wesley, 1973, Ch. 1.2.

Appendix A. An example appendix

Authors including an appendix section should do so after References section. Multiple appendices should all have headings in the style used above. They will automatically be ordered A, B, C etc.

Appendix A.1. Example of a sub-heading within an appendix

There is also the option to include a subheading within the Appendix if you wish.