Is deep learning better than kernel regression for functional connectivity prediction of fluid intelligence?

Tong He[1,2], Ru Kong[1,2], Avram J.Holmes[3], Mert R. Sabuncu[4], Simon B. Eickhoff[6,7], Danilo Bzdok[5], Jiashi Feng[2], B.T. Thomas Yeo[1,2,8,9,10]
[1] Clinical Imaging Research Centre Singapore Institute for Neurotechnology, National University of Singapore, Singapore
Email: thomas.yeo@nus.edu.sg
[2] Department of Electrical and Computer Engineering, National University of Singapore, Singapore
[3] Department of Psychology, Yale University, New Haven, CT, USA
[4] School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, USA
[5] Department of Psychiatry, Psychotherapy and Psychosomatics, RWTH Aachen University, Germany
[6] Institute of Systems Neuroscience, Medical Faculty, Heinrich Heine University Düsseldorf, Düsseldorf, Germany
[7] Institute of Neuroscience and Medicine, Brain & Behaviour (INM-7), Research Centre Jülich, Jülich, Germany
[8] Martinos Center for Biomedical Imaging, Massachusetts General Hospital, Charlestown, MA, USA
[9] Centre for Cognitive Neuroscience, Duke-NUS Medical School, Singapore
[10] NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore

***Abstract*—In recent years, deep learning has transformed the field of machine learning. In the field of neuroimaging, there are increasing interests in adopting deep learning techniques. However, deep neural networks (DNNs) usually need large quantity of data to perform well, which is often lacking in neuroimaging. In this work, we evaluated three different DNNs (fully-connected neural network, BrainNetCNN** [1]**, and graph convolutional neural network** [2]**) for functional connectivity (FC)-based prediction of fluid intelligence using the Human Connectome Project. These DNNs were compared with kernel regression, a classical machine learning algorithm. Our results suggested that the DNNs did not outperform kernel regression. However, we do not preclude the possibility that with more participants or different FC features, DNNs might eventually outperform their classical counterpart.**

Keywords—Deep learning, fMRI, neuroimaging, Functional Connectivity, machine learning, kernel regression

I. Introduction

Deep neural networks (DNN) have been used to achieve many state-of-the-art results in multiple areas in machine learning [3], [4]. Therefore, there has been a significant interest in applying DNNs to neuroimaging. For example, DNNs have been used for image classification and segmentation. Various types of DNNs are used to predict disorder using functional Magnetic Resonance Imaging (fMRI) input [5]. Convolutional neural network (CNN), a type of DNN, is widely used to perform lesion/tumor detection and segmentation tasks [6], [7]. DNN is also applied to image construction/enhancement, for example, researchers are using DNNs to perform imaging modality conversion [8], [9].

However, training DNN from scratch usually requires large scale dataset, e.g. ImageNet [3] (14 million images). This amount of data is lacking in neuroimaging today. Therefore, we hypothesize that in certain neuroimaging applications, deep learning might not necessarily outperform classical machine learning approaches. In this work, we compare DNN approaches and classical machine learning for functional connectivity (FC)-based behavioral prediction.

Resting state functional magnetic resonance imaging (rs-fMRI) functional connectivity reflects the synchrony of fMRI signals between brain regions [10], [11]. Functional connectivity is often defined as the Pearson correlation coefficients between time courses of two regions. Given a parcellation scheme, one can define a FC matrix by computing the correlations between all pairs of parcels. The FC matrix is able to act as an identifying fingerprint of different person and can be also used to predict some fundamental cognitive traits, such as fluid intelligence [12], [13]. Since fluid intelligence reflects inherent cognitive ability, there is great interest in analyzing its brain basis.

Here, we compare three established DNNs with the classical machine learning algorithm, kernel regression to predict fluid intelligence using FC. Two of the DNN approaches were proposed for neuroimaging applications, and have previously served to predict infant age, neurodevelopmental outcome [1] and disease status [2]. We apply these algorithms to the Human Connectome Project (HCP) dataset [14] to predict fluid intelligence. Experiments suggest that, in this context, the DNNs do not outperform the classical machine learning algorithm.

II. Experimental setup

In this section, we explain the FC-based behavioral prediction setup and the various algorithms that we compared in a systematic fashion.

A. Basic setup

We assumed that there are M_{total} subjects in a dataset. Each subject i has a $N \times N$ FC matrix c_i, as well as fluid intelligence y_i. The goal was to predict fluid intelligence using the FC matrix. We compared four algorithms, including three DNNs and one classical machine learning algorithm. The performance was evaluated by the correlation between the predicted and the real fluid intelligence measure across subjects.

B. Dataset and data preprocessing

For input data, we considered rs-fMRI data from 954 HCP subjects [15]. The fMRI data were first cleaned by ICA-FIX [15]. Then, we applied motion censoring (volumes with FD>0.2mm or DVARS>75) and global signal regression to the denoised data. Finally, FC was computed among 400 cortical [16] and 19 subcortical ROIs, resulting in a 419 x 419 FC matrix per subject, which is the input data.

978-1-5386-6859-7/18/$31.00 ©2018 IEEE

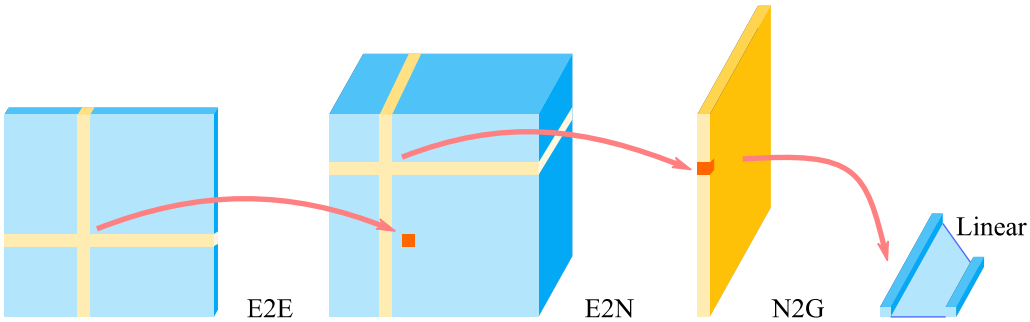

Fig. 1. Structure of BrainNetCNN [1], This network takes in a FC matrix as input and outputs behavior measures. BrainNetCNN consists of four different types of layers, Edge-to-Edge (E2E) layer, Edge-to-Node (E2N) layer, Node-to-Graph (N2G) layer, and fully connected (Linear) layer.

The output data is the fluid intelligence measure, represented by PMAT24_A_CR [17] (Penn Progressive Matrices: number of correct responses) in HCP dataset.

C. Cross-validation procedure

We performed 20-fold cross-validation. The fold split respected the family structure of subjects. Subjects from the same family were kept in same fold. Each model was trained on the 19 data splits and used to predict behavior on the subjects in the test split. We regressed out age, gender and motion from the fluid intelligence measure for the training folds, and then we applied the same regression on the test fold using the regression coefficients from training folds. Hyperparameters were selected via inner-loop cross-validation by performing training and validation inside 19 splits. For hyperparameters, see details in the next section for the different algorithms.

D. Corrected resampled t-test

In order to compare the cross-validation performance of these algorithms, we performed corrected resampled t test for 20-fold cross-validation [18]. This t-test is performed on the correlation results of 20 test data splits. This t-test corrects the variance of 20-fold cross-validation, because the sample variance across folds underestimates the true variance.

III. Methods

A. Kernel Regression

Kernel regression [19], [20] is a classical machine learning algorithm which can capture higher-order non-linear structure. Kernel regression can be utilized to predict fluid intelligence measure y with kernel matrix $K(\cdot,\cdot)$. Motivated by [12], the i-th row, j-th column of K is defined to be the correlation between the vectorized FC matrix of the i-th and j-th subject. The fluid intelligence measure y_i of subject i can be represented as $y_i = h(c_i) + e_i$, where e_i is a noise term, and $h(c_i)$ is a nonparametric function of vectorized FC c_i. $h(c_i)$ can be expressed as $h(c_i) = \sum_{j=1}^{M} \alpha_j K(c_j, c_i)$, where α is an unknown weight and M is the total number of training subjects. In order to estimate α, we maximize the penalized likelihood function:

$$J(h) = -\frac{1}{2}\sum_{i=1}^{M}\{y_i - h(c_i)\}^2 - \frac{1}{2}\lambda\|h\|^2 \qquad (1)$$

where λ is a hyperparameter which controls the l_2 regularization term. Equation (1) can also be written as:

$$\boldsymbol{\alpha} = \underset{\alpha}{\operatorname{argmin}} \frac{1}{2}(\boldsymbol{y} - \mathbb{K}\boldsymbol{\alpha})^T(\boldsymbol{y} - \mathbb{K}\boldsymbol{\alpha}) + \frac{\lambda}{2}\boldsymbol{\alpha}^T\mathbb{K}\boldsymbol{\alpha}. \qquad (2)$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_M]^T$, $\boldsymbol{y} = [y_1, y_2, \dots, y_M]^T$, and $\mathbb{K}$ is the $M \times M$ training subjects FC similarity. A wide range of λ were tested in the inner-loop cross-validation, and the λ with best performance was applied to the test fold. With the optimized $\boldsymbol{\alpha}$ and λ, the behavior measure y_x of subject x can be estimated by $h(c_x)$. For a detailed explanation, please refer to [20].

B. Fully-connected Neural Network (FNN)

Fully-connected Neural Network (FNN) [21] or multilayer perceptron is a class of feedforward neural network. This architecture takes vector as input and output. FNN consists of several layers of neurons (nodes). All the nodes of each layer are connected to the all the nodes of the next and previous layer except the input and output layer, which is only connected to the inner layer of the network. For all the intermediate layers, every node has a nonlinear activation function, like ReLU (*f(x)=max(0,x)*).

In this work, it takes in vectorized FC as input and outputs fluid intelligence measure. The number of layers, number of nodes in layers, rate of dropout, types of activation, regularization and optimizer have been tuned during hyperparameter tuning for inner-loop cross-validation.

C. BrainNetCNN

BrainNetCNN [1] is a specially designed CNN for FC. The structure of BrainNetCNN is shown in Fig. 1. BrainNetCNN takes in original 2D FC matrix directly as input. This network has been used to predict infant age and neurodevelopmental outcome [1]. Convolution operation is performed on rows and columns instead of patches of FC matrix. BrainNetCNN has 3 different types of special designed layers, Edge-to-Edge (E2E) layer, Edge-to-Node (E2N) layer, and Node-to-Graph (N2G) layer. E2E layer convolves on the entire *i*-th row and *j*-th column for each nodes n_{ij}. Intuitively, it is a summation of 1D convolution on row and column. Since the *i*-th row or column elements in FC matrix are the correlations between *i*-th ROI and all ROIs in brain, the E2E layer convolves all the FC correlation related to ROI *i* and *j*. E2N layer performs the same convolution only on the diagonal nodes n_{ii}. N2G layer is just a fully connected layer. A normal fully-connected (linear) layer is at the end of the BrainNetCNN.

In this work, we used FC matrix directly as input to predict fluid intelligence. In cross-validation process, we tuned the filters number in E2E, E2N and N2G layers, rate of dropout, types of activation, regularization and optimizer in inner-loop cross-validation.

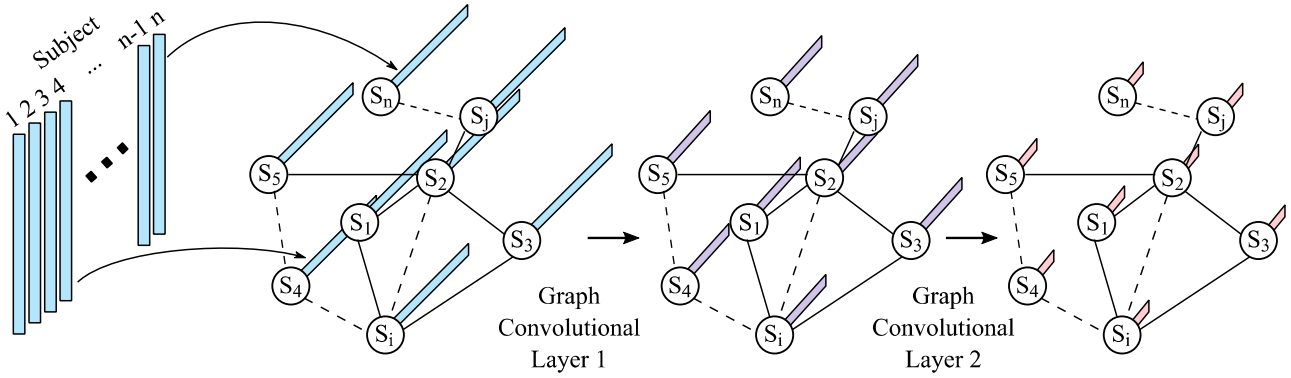

Fig. 2. Structure of GCNN [2]. One node represent one subject. It takes in a vectorized FC of all subjects as input and outputs behavior measures of all subjects.

D. Graph Convolution Neural Network (GCNN)

Graph Convolution Neural Network (GCNN) [2] is a special CNN based on graph convolution, and its core layer is called graph convolutional layer. GCNN takes in vectorized FC of all subjects as input at same time. Since every layer actually incorporates a graph with subject as vertex (node), the vectorized FC of each subject is passed to each node respectively. The graph edges are determined by the correlation between subjects' FC and are filtered by a tuned threshold. The convolution in the graph convolutional layer can be seen as a graph spectral filter based on estimated graph Laplacian. Moreover, GCNN does not have max-pooling layer, since each node represent different subject. For detail explanation, please check [2].

GCNN has been used to predict disease status of autism and Alzheimer's disease [2]. In this work, we are using vectorized FC to predict fluid intelligence. During cross-validation process, we tuned many hyperparameters, including graph setup, method of graph Laplacian estimation, numbers of graph convolutional layer, numbers of filters of graph convolutional layer, and regularization.

IV. Results

We compared the three deep neural networks, FNN, BrainNetCNN, and GCNN, against the kernel regression in the fluid intelligence prediction of HCP dataset.

A. Deep neural network implementation

Based on the Github code provided by [1], [22], we implement three DNNs with Keras [23] framework on NVIDIA Titan Xp GPU using CUDA. Our implementation successfully reproduced the results on the datasets provided in their Github. Then, we carefully tuned three DNNs and kernel regression with 20-fold cross-validation. Here are the details on our implementation and final architecture.

- For all three DNNs, the fluid intelligence data is z-normalized. We used mean squared error (MSE) as loss function and Adam [24] as optimizer.
- FNN consists of 4 fully-connected layers with 256, 96, 32 and 1 nodes. FNN has dropout of 0.6 and L2 regularization of 0.02.
- BrainNetCNN consists of 4 layers: E2E layer with 16 filters, E2N layer with 64 filters, N2G layer with 30 filters, and fully connected layer with 1 node. A dropout of 0.5 is added between E2N and N2G layer. The activation is using LeakyReLU [25] with 0.33 alpha.
- GCNN consists of 2 graph convolutional layers with numbers of nodes of 16 and 1. Dropout rate of 0.3 and a L2 regularization of 5e-4 are used in the GCNN. The following types of edges are kept for graph construction: 1) edges with top 5% correlation 2) top 5 correlation edges for each vertex. The graph convolution filter is estimated by Chebyshev polynomial of degree 8.

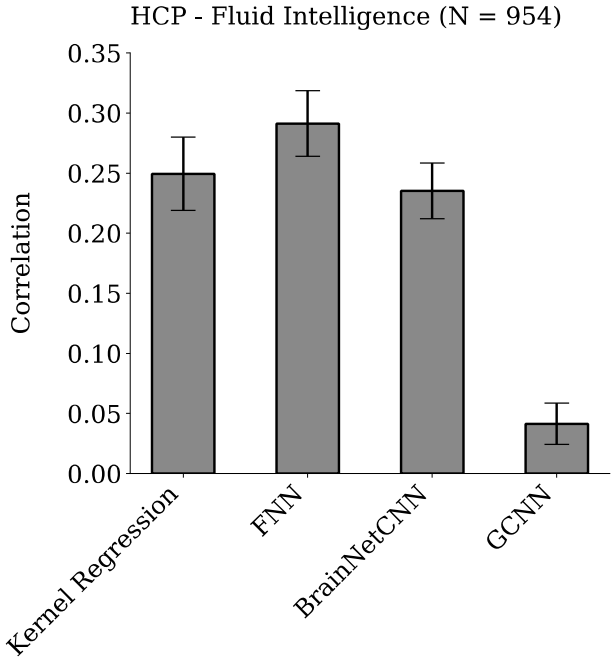

Fig. 3. 20-fold cross-validation accuracy (correlation) of three DNNs and kernel regression on HCP dataset FC-based fluid intelligence prediction. The error bar represents standard error.

B. Fluid intelligence prediction

The results of FC-based fluid intelligence prediction are shown in Fig. 3, including the cross-validation correlation and standard error for each model.

We found that both specially designed neural networks, BrainNetCNN and GCNN, did not outperform the kernel

regression. Only FNN, a classical neural network, has shown nominally higher correlation than the kernel regression. However, there is no statistical difference ($p = 0.1050$, corrected resampled t-test) between the performance of kernel regression and FNN. In summary, no DNNs outperformed kernel regression.

For DNNs, two DNNs, BrainNetCNN and GCNN, were developed and designed specifically for neuroimaging data, i.e., to exploit the structure of a FC matrix or information from whole population. However, a generic FNN has the highest correlation in our experiment. BrainNetCNN has a comparable performance compared to kernel regression and FNN, while GCNN does not perform very well in our experiment.

V. Conclusion

In summary, we tested three DNNs against classical machine learning that can capture higher-order non-linear strcture in FC-based fluid intelligence prediction. In our experiments, DNNs and kernel regression yielded similar performance except GCNN when predicting fluid intelligence from FC for HCP dataset. Given the well-known difficulties in tuning DNNs, it is possible that better tuning could yield better performance for the DNNs. However, our results are consistent with the broader deep learning literature that large quantity of data are necessary for deep learning to be more effective than classical machine learning techniques.

Future work will focus on the comparison of classical machine learning and deep learning algorithms on more behavior measures and dataset with larger population, including finding a population threshold that deep learning requires to outperform classical machine learning algorithm.

Acknowledgment

This work was supported by Singapore MOE Tier 2 (MOE2014-T2-2-016), NUS Strategic Research (DPRT/944/09/14), NUS SOM Aspiration Fund (R185000271720), Singapore NMRC (CBRG/0088/2015), NUS YIA and the Singapore National Research Foundation (NRF) Fellowship (Class of 2017). Our research also utilized resources provided by the Center for Functional Neuroimaging Technologies, P41EB015896 and instruments supported by 1S10RR023401, 1S10RR019307, and 1S10RR023043 from the Athinoula A. Martinos Center for Biomedical Imaging at the Massachusetts General Hospital. Our computational work was partially performed on resources of the National Supercomputing Centre, Singapore (https://www.nscc.sg). Data were provided by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University. The Titan Xp used for this research was donated by the NVIDIA Corporation.

References

[1] J. Kawahara *et al.*, "BrainNetCNN: Convolutional neural networks for brain networks; towards predicting neurodevelopment," *Neuroimage*, vol. 146, pp. 1038–1049, 2017.

[2] S. Parisot *et al.*, "Spectral Graph Convolutions for Population-Based Disease Prediction," in *Medical Image Computing and Computer-Assisted Intervention − MICCAI 2017*, 2017, pp. 177–185.

[3] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[4] Y. Goldberg, *Neural Network Methods for Natural Language Processing (Synthesis Lectures on Human Language Technologies)*, vol. 10, no. 1. 2017.

[5] S. Vieira, W. H. L. Pinaya, and A. Mechelli, "Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications," *Neurosci. Biobehav. Rev.*, vol. 74, pp. 58–75, 2017.

[6] K. Kamnitsas *et al.*, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, 2017.

[7] T. Brosch, L. Y. W. Tang, Y. Yoo, D. K. B. Li, A. Traboulsee, and R. Tam, "Deep 3D Convolutional Encoder Networks With Shortcuts for Multiscale Feature Integration Applied to Multiple Sclerosis Lesion Segmentation," *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1229–1239, 2016.

[8] K. Bahrami, F. Shi, I. Rekik, and D. Shen, "Convolutional Neural Network for Reconstruction of 7T-like Images from 3T MRI Using Appearance and Anatomical Features," in *MICCAI 2016 DL workshop*, vol. 10008, no. October, 2016, pp. 39–47.

[9] V. D. Calhoun, M. F. Amin, D. Hjelm, E. Damaraju, and S. M. Plis, "A deep-learning approach to translate between brain structure and functional connectivity," *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 6155–6159, 2017.

[10] B. Biswal, Y. FZ, H. VM, and H. JS, "Functional connectivity in the motor cortex of resting human brain using echo-planar MRI," *Magn Reson Med*, vol. 34, no. 9, pp. 537–541, 1995.

[11] R. L. Buckner, F. M. Krienen, and B. T. T. Yeo, "Opportunities and limitations of intrinsic functional connectivity MRI," *Nat. Neurosci.*, vol. 16, no. 7, pp. 832–837, 2013.

[12] E. S. Finn *et al.*, "Functional connectome fingerprinting: Identifying individuals using patterns of brain connectivity," *Nat. Neurosci.*, vol. 18, no. 11, pp. 1664–1671, 2015.

[13] B. T. T. Yeo, J. Tandi, and M. W. L. Chee, "Functional connectivity during rested wakefulness predicts vulnerability to sleep deprivation," *Neuroimage*, vol. 111, pp. 147–158, 2015.

[14] D. C. Van Essen, S. M. Smith, D. M. Barch, T. E. J. Behrens, E. Yacoub, and K. Ugurbil, "The WU-Minn Human Connectome Project: An overview," *Neuroimage*, vol. 80, pp. 62–79, 2013.

[15] S. M. Smith *et al.*, "Resting-state fMRI in the Human Connectome Project," *Neuroimage*, vol. 80, pp. 144–168, 2013.

[16] A. Schaefer *et al.*, "Local-Global Parcellation of the Human Cerebral Cortex from Intrinsic Functional Connectivity MRI," *Cereb. Cortex*, no. March 2018, pp. 1–20, 2017.

[17] W. B. Bilker, J. A. Hansen, C. M. Brensinger, J. Richard, R. E. Gur, and R. C. Gur, "Development of Abbreviated Nine-Item Forms of the Raven's Standard Progressive Matrices Test," *Assessment*, vol. 19, no. 3, pp. 354–369, 2012.

[18] R. R. Bouckaert and E. Frank, "Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms," *Adv. Knowl. Discov. data Min.*, pp. 3–12, 2004.

[19] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. 2012.

[20] R. Kong *et al.*, "Spatial Topography of Individual-Specific Cortical Networks Predicts Human Cognition, Personality and Emotion," *bioRxiv*, p. 213041, 2018.

[21] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[22] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," *Int. Conf. Learn. Represent.*, pp. 1–14, 2017.

[23] F. Chollet, "Keras," *GitHub*, 2015. [Online]. Available: https://github.com/fchollet/keras.

[24] D. P. Kingma and J. L. Ba, "Adam: a Method for Stochastic Optimization," *Int. Conf. Learn. Represent. 2015*, pp. 1–15, 2015.

[25] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," *Proc. 30 th Int. Conf. Mach. Learn.*, vol. 28, p. 6, 2013.