International Journal of application or Innovation in engineering and management

Ijaiem.com/Aug 2024/ Volume 13/Issue 2/Article No-1/35-41

ISSN: 2319-4847

DEEP NETWORK OPTIMIZATION UTILIZING ADAPTIVE RATES

Vanapamula Veerabrahmachari[1] , Mr. Merugu Anand Kumar[2], Nagam Aanjaneyulu[3], Gudipati Mohan Singh Yadav[4]

Associate Professor[1,4], Associate Professor[2,3]

vveerabrahmachari@gmail.com[1] ,meruguanand502@gmail.com[2],
anji.amrexamcell@gmail.com[3],gudipatimohan20@gmail.com[4]

Department of CSE, A.M. Reddy Memorial College of Engineering and Technology, Petlurivaripalem, Narasaraopet, Andhra Pradesh

Article Info

Received: 29-07-2024 Revised: 06 -08-2024 Accepted: 16-08-2024 Published:28/08/2024

Abstract: - Profound learning structures are turning out to be more confounded, bringing about weeks, if not months, of tutoring time. This drowsy schooling is brought about by "evaporating inclinations," in which the angles utilized by engendering are gigantic for loads interfacing profound (layers close to the yield layer) and little for loads associating shallow (layers close to the information layer), bringing about sluggish learning inside the shallow layers. Besides, low arch seat factors have been displayed to create during non-raised illnesses, like profound neural organizations, which essentially eases back learning [1]. In this paper, we present an advancement technique for profound neural organization training that plans to tackle the two issues referenced above by utilizing study costs that are explicit to each layer in the organization and versatile to the ebb and flow of the element, permitting us to foster burden information at low curve components. This empowers us to learn quicker in the organization's shallow layers and break out extreme mistakes of low shape saddle parts in a short measure of time. We utilize our procedure to huge picture gloriousness datasets like as MNIST, CIFAR10, and Image Net, and exhibit that it further develops exactness while diminishing the measure of time required for preparing over immense strategies.

I. INTRODUCTION

Profound neural organizations have demonstrated to be exceptionally effective lately, accomplishing cutting edge results on a scope of errands, for example, picture grouping [2], face acknowledgment [3], feeling investigation [4], voice acknowledgment [5], etc. A typical inclination can be found in these articles: as the measure of preparing information increments, so does the intricacy of the profound organization engineering. Notwithstanding, even with superior equipment, preparing progressively complex profound organizations might require weeks or months. Therefore, more remarkable techniques are needed for preparing profound organizations. Profound neural organizations learn significant level qualities by executing a progression of non-direct activities. Leave An alone a preparation informational index with n information focuses a1, a2, and x Mr, just as related marks B = bi n i=1. Expect that f is the initiation job of a 3-layer organization. Allow X1 and X2 to address the loads we're attempting to learn on the - line, i.e., X1 connotes the loads between the first and second layer hubs, and X2 implies the loads between the second and third layer hubs. For this model, the learning issue might be expressed as the accompanying streamlining issue:

The enactment work f, which is normally a sigmoid or tan capacity, might be any non-direct planning. Amended direct (Relook) units (f (z) =max 0, z) have as of late been well known since they appear to be not difficult to prepare and give better results to specific issues [6]. The non-raised objective (1) is commonly brought down by utilizing iterative techniques, (for example, back-spread) determined to combine to a reasonable nearby least. Most iterative techniques bring about added substance adjustments to the shape's boundary set x (in our case, weight networks).

Where x (k) is a very much picked alteration. Note that we utilize a fairly unique documentation here than in customary enhancement writing, in that we coordinate the stage size or learning rate t (k) into x. (k). This is done to make it simpler to talk about different streamlining techniques in the following areas. Accordingly, in the boundaries, x (k) shows the update and is comprised of a mission course and a stage size or learning rate t (k), which decides how enormous a stage toward that path ought to be taken. The most well-known refreshing standards are slope plunge variations, in which the hunt heading is given by the negative angle g (k):

The inclination can't be precisely estimated since the preparation information for these profound organizations generally comprises of millions or billions of information focuses. All things considered, the angle is constantly processed utilizing a solitary information point or few information focuses. This is the reason for stochastic slope drop (SGD), the most generally utilized technique for profound net arrangement [7]. SGD should pick an underlying learning rate physically, then, at that point develop a learning rate update law that diminishes it over the long run (for instance, outstanding rot with time). SGD's yield, then again, is exceptionally delicate to this update choice, driving in versatile strategies that consequently modify the learning rate as the machine learns [8], [9]. As these plummet techniques are used to prepare profound organizations, new issues arise. As the quantity of layers in an organization builds, the inclinations that are communicated back to the underlying layers become minuscule. This considerably lessens the pace of learning in the early layers, just as the general organization combination [10].

For high-dimensional non-arched subjects like profound organizations, it has as of late been shown that the event of neighborhood minima with critical incorrectness comparative with the worldwide least is dramatically little in the quantity of measurements. All things being equal, these issues incorporate a dramatically enormous number of low-ebb and flow high-blunder saddle spots [1], [11], [12]. By inspecting the pathways of negative ebb and flow, inclination drop strategies ordinarily disappear from saddle focuses. Because of the helpless curve of minuscule negative eigenvalues, nonetheless, the developments made become very limited, deferring adapting fundamentally. In this article, we propose a strategy for tending to the previously mentioned issues. The principle commitment of our system is expressed here.

• Each substrate in the organization has its own learning rate. To make up for the confined size of inclinations in shallow layers, quicker learning rates are required.

• Learning rates for each layer start to increment at low ebb and flow focuses. This permits the procedure to promptly keep away from high-mistake, low-curve saddle spots, which are plentiful in profound organizations.

• It works with most contemporary stochastic inclination streamlining methods that use a worldwide learning scale.

• Compared to customary stochastic inclination procedures, it requires next to no additional handling and needn't bother with any extra putting away of past angles, as AdaGrad [9] does.

• In Section II, we go through a few mainstream inclination strategies that have functioned admirably for profound organizations. In Section III, we characterize our enhancement technique. At long last, in Section IV, we contrast our methodology with regular advancement procedures on datasets like MNIST, CIFAR10, and Image Net.

II. Associated WORK

SGD (Stochastic Gradient Descent) is perhaps the most broadly utilized huge scope AI strategies, inferable from its simplicity of execution. The boundary refreshes in SGD are characterized by conditions (2) and (3), and the learning rate diminishes over the long run as emphasizes approach a nearby ideal. The learning rate is refreshed consistently.

In the event that the client picks the underlying learning rate t (0) and the learning rate p. Numerous improvements to the essential slope drop technique have been recommended. Newton's strategy, which

International Journal of application or Innovation in engineering and management

Ijaiem.com/Aug 2024/ Volume 13/Issue 2/Article No-1/35-41

ISSN: 2319-4847

ascertains the stage scale utilizing the Hessian of the target work f(x), is an unmistakable methodology in the curved advancement writing:

Sadly, as the quantity of components increments, ascertaining the Hessian turns out to be amazingly computationally costly, even at a humble scope. Subsequently, different changes have been suggested that endeavor to either enhance the utilization of first-request data or gauge the Hessian target work. In this exposition, we center around first-request approach changes. The old style energy procedure [13] is a technique that expands the learning rate for boundaries where the slope continually focuses a similar way while bringing down the learning rate for boundaries where the angle changes rapidly. For an outstanding rot, the update condition monitors past boundary changes:

The force coefficient is alluded to as?? [0, 1], and the worldwide learning rate is t > 0. In specific occasions, Nesterov's Accelerated Gradient (NAG) [14], a first-request measure, beats angle drop as far as union rate. This technique predicts the inclination for the following emphasis and changes the learning rate for the current cycle dependent on the anticipated slope. Accordingly, if the slope for the accompanying stage is more prominent, the current cycle's learning rate will increment, however in case it is lower, it will dial back. [15] as of late shown that this strategy might be thought about as a force technique utilizing the change condition:

At the point when utilized on profound organizations [15], this strategy will arrive at extraordinary degrees of effectiveness by using an appropriately planned irregular instatement and a specific sort of leisurely expanding plan for. Late examination has shown that using a learning rate explicit to every boundary, as opposed to a typical learning rate for all boundaries, might be a substantially more productive methodology. AdaGrad [9] is a notable apparatus that utilizes the accompanying updating rule:

$$\Delta x^{(k)} = -\frac{t}{\sqrt{\sum_{i=1}^{k}(g^{(i)})^2}} g^{(k)} \qquad (8)$$

The l2 standard is the denominator of the relative multitude of angles from past emphasess. This builds the worldwide learning rate t, which is shared by all boundaries, to give a boundary explicit learning rate. One burden of AdaGrad is that it gathers angles from past cycles, the amount of which will in general increment all through arrangement. This diminishes the quantity of compelling preparing emphasess by diminishing the learning rate on every boundary (alongside weight rot) until each is imperceptibly little. AdaDelta [8] is a strategy dependent on AdaGrad that attempts to settle a portion of the issues referenced previously. AdaDelta gathers the angles in going before time estimations utilizing a dramatically rotting normal of the squared inclinations. This keeps the denominator from turning out to be imperceptibly little and guarantees that the boundaries are changed even after countless reiterations. It additionally replaces the worldwide learning rate t with a dramatically declining amount of the squares of the boundary changes x across the first cycles. This technique has been demonstrated to perform genuinely well when used to prepare profound organizations, and is considerably less delicate to hyper-boundary determination. Notwithstanding, it misses the mark concerning different techniques like SGD and AdaGrad as far as exactness [8].

I. OUR METHOD

"Due to the "evaporating angles" marvel, shallow organization layers appear to have significantly more modest inclinations than profound levels, once in a while changing arranged by extent starting with one layer then onto the next [10]." In many past work on improvement for profound organizations, techniques either use a worldwide learning rate that is copied across all boundaries or utilize a versatile learning rate that is extraordinary to every boundary. Our methodology takes utilization of the way that boundaries in a similar layer have comparative inclination sizes and hence may successfully share a learning rate. Layer-explicit learning rates might be utilized to speed up layers with more modest slopes. Another advantage of this methodology is that it keeps our framework computationally productive by staying away from the calculation of countless boundary explicit learning speeds. At last, as referenced in Section I, we need our strategy to make huge strides at low curve focuses to abstain from getting the hang of being eased back at high-mistake low ebb and flow saddle spots. Let t (k) be the learning rate at the k-th cycle for any normal streamlining method. On account of SGD, this would be given by condition 4, while with AdaGrad, it would just be the worldwide learning rate t, as in condition 8. We suggest that t (k) be changed to:

International Journal of application or Innovation in engineering and management

Ijaiem.com/Aug 2024/ Volume 13/Issue 2/Article No-1/35-41

ISSN: 2319-4847

$$t_l^{(k)} = t^{(k)}(1 + \log(1 + 1/(\|g_l^{(k)}\|_2))) \qquad (9)$$

g (k) l shows the vector of the boundary inclinations at the k-the emphasis in the l-the layer, though t (k) l indicates the new learning rate for the boundaries at the k-the cycle in the l-the layer. Thus, we can see that we just use inclinations from a similar layer to register the learning rate for that line. It's additionally worth recalling that, in contrast to prior versions, we don't use any inclinations, which saves space. At the point when the inclinations in a layer are amazingly enormous, the condition essentially improves to utilizing the standard learning rate t (k), as displayed in condition 9. Notwithstanding, we are bound to be in a low bend point with incredibly unobtrusive slants. Subsequently, the condition expands the learning rate to ensure that the organization's initial layers learn quicker and that high-mistake low-shape saddle spots are handily stayed away from. We might use this layer-explicit learning rate notwithstanding SGD. In such occurrence, utilizing condition 3, the change will be:

$$\Delta x_l^{(k)} = -t_l^{(k)} g_l^{(k)} \qquad (10)$$

$$= -t^{(k)}(1 + \log(1 + 1/(\|g_l^{(k)}\|_2)))g_l^{(k)} \qquad (11)$$

Where (k) l denotes the change at the k-the iteration in the l-the layer parameters. Similarly, to use our updated learning speeds, we should change AdaGrad's upgrade equation (8).

$$\Delta x_l^{(k)} = -\frac{t_l^{(k)}}{\sqrt{\sum_{i=1}^{k}(g_l^{(i)})^2}} g_l^{(k)} \qquad (12)$$

In contrast to AdaGrad, which utilizes an alternate learning rate for every boundary, we use a solitary learning rate for each layer that is shared by all loads in that layer. Moreover, AdaGrad changes the learning rate dependent on the full foundation of angles seen for that weight, while we essentially adjust the learning pace of a layer dependent on inclinations saw in the current cycle for all loads in that layer. Thus, our methodology disallows the assortment of angle data from prior cycles just as the estimation of learning rates for every boundary, making it less computationally and memory requesting than AdaGrad. The proposed layer extraordinary learning rates function admirably for huge scope datasets like Image Net (when reached out over SGD), while AdaGrad neglects to unite to a decent arrangement. For the proposed strategy, any current enhancement procedure that uses a worldwide learning rate, has a layer-explicit learning rate, and promptly gets away from saddle spots without forfeiting calculation or memory utilization might be used. On standard datasets, utilizing our versatile learning rates on top of known enhancement strategies almost perpetually further develops productivity, as we show in Section IV. The proposed technique might be applied with any current enhancement procedure that uses a worldwide learning rate. This empowers for a layer-explicit learning rate to be accomplished, just as a decrease in computational expenses, which assists with staying away from saddle spots sooner. On customary datasets, utilizing our versatile learning rates on top of known enhancement techniques almost perpetually further develops effectiveness, as we show in Section IV.

Aftereffects OF EXPERIMENTATION

A. Dataset

We show picture order results for three ordinary datasets: MNIST, CIFAR10, and Image Net (ILSVRC 2012 dataset, a piece of the Image Net test). 60,000 advanced written by hand pictures for readiness and 10,000 computerized transcribed pictures for study are accessible from MNIST. CIFAR10 is comprised of ten gatherings of 6,000 pictures in each class. Picture Net uses 1.2 million tone photographs from 1000 unique associations. B. Experimentation Data We use Cafe [16] to implement our technique. Bistro gives enhancement methods like Stochastic Gradient Descent (SGD), Nester's Accelerated Gradient (NAG), and AdaGrad. For a reasonable examination of best in class strategies, we utilize our versatile layer-explicit learning rate approach on top of both of these advancement techniques. Our examinations show the adequacy of our technique on convolutional neural organizations on three datasets. We apply a similar worldwide learning rate on CIFAR10 that we use in Cafe. Albeit, in contrast with past advancement procedures, our technique regularly builds the layer-explicit learning rate dependent on the worldwide learning rate, we start with a little lower learning pace of 0.006 to make the Image Net examination less brutal for learning. SGD was begun utilizing the learning rate portrayed in [2] for Image Net examination. 1) MNIST: We use a similar plan as Lent for our MNIST tests. On the MNIST dataset, we show the consequences of utilizing our proposed layer-explicit learning rates

International Journal of application or Innovation in engineering and management

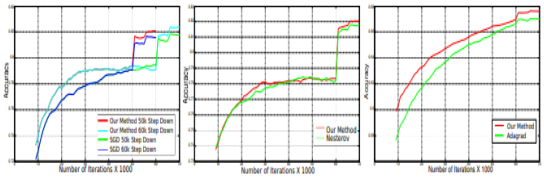
Ijaiem.com/Aug 2024/ Volume 13/Issue 2/Article No-1/35-41

ISSN: 2319-4847

notwithstanding stochastic slope plummet, Nester's sped up inclination strategy, and Adored. We just show the exactness and misfortune for the initial 2,000 emphasess since all techniques promptly concur on this dataset. I will show you a table.

Iteration	SGD	Ours-SGD	Nesterov	Ours-NAG	AdaGrad	Ours-AdaGrad
200	7.90 ± 0.44	7.25 ± 0.46	6.66 ± 0.47	5.37 ± 0.5	4.12 ± 0.32	3.40 ± 0.3
600	3.28 ± 0.22	3.05 ± 0.21	3.01 ± 0.19	2.84 ± 0.17	2.21 ± 0.14	1.95 ± 0.16
1000	1.89 ± 0.08	1.80 ± 0.13	1.92 ± 0.07	1.83 ± 0.18	1.68 ± 0.11	1.57 ± 0.14
1400	1.60 ± 0.11	1.49 ± 0.09	1.74 ± 0.12	1.52 ± 0.11	1.61 ± 0.08	1.61 ± 0.09
1800	1.52 ± 0.09	1.41 ± 0.12	1.56 ± 0.09	1.37 ± 0.09	1.41 ± 0.09	1.34 ± 0.08

TABLE me: After many iterations for stochastic gradient descent, the accelerated gradient of Nester and Adored with their layer-specific adaptive models, the mean error rate on MNIST as shown in the table. Each process was run ten times, and the mean and standard deviation were calculated.



(a) Stochastic Gradient Descent (b) Nesterov's Accelerated Gradient (c) AdaGrad

Fig. Fig. 1: CIFAR data set: accuracy-showing plots (Figures 1a-1c) contrasting SGD, NAG and AdaGrad, each with our layer-wise adaptive learning speeds. We display results for the SGD plot both when we move down the learning rate at 50,000 iterations and at 60,000 iterations.

The mean precision and standard deviation were determined after every activity was rehashed multiple times. Our proposed layer-explicit learning rate is reliably more noteworthy than Nesterov's sped up inclination, stochastic angle plunge, and AdaGrad. The proposed technique, which incorporates stochastic angle plummet, Nesterov's sped up slope, and AdaGrad, likewise gets the best precision of 99.2 percent in the entirety of the tests.

2) CIFAR10 (Conference on International Food Aid Regulations):

On CIFAR10, we utilize a convolutional neural organization with two layers of 32 trademark maps comprised of 5 to 5 convolution portions, each with 3 to 3 all out pooling layers. From that point onward, we have another convolution sheet with 64 capacities mappings from a 5?? 5 convolution portions, just as a 3?? 3 max pooling layer. At last, we have a totally associated layer with 10 mystery hubs and a delicate max strategic relapse layer. After every convolution sheet, a ReLu non-linearity is added. This design is indistinguishable from that portrayed by Cafe. The learning execution was 0.001 during the initial 60,000 emphasess, and it was diminished by a factor of ten at 60,000 and 65,000 cycles. On this dataset, we find that our methodology reliably has lower last blunder and disappointment than SGD, NAG, and AdaGrad (Table II). After stage down, our versatile methodology yields more unfortunate precision than both SGD and NAG. Utilizing our advancement strategy, we can accomplish a 0.32 percent improvement in SGD precision over the mean exactness (without altering the organization engineering). Despite the fact that we lessen the learning rate after 50,000 cycles (taking 60000 altogether), we acquire a precision of 82.08 percent, which is more noteworthy than SGD after 70,000 emphasess, essentially decreasing the necessary preparing time Fig. 1. Since our technique joins a lot quicker when joined with SGD, the learning rate stage down might be finished impressively sooner, conceivably decreasing preparing time much further. While Adagrad doesn't perform well with default settings on CIFAR10, it shows a 1.3 percent improvement in normal end exactness, just as a huge decrease in preparing time.

Picture Net (#3):

We use an execution of Alex Net [2] in Cafe, profound convolutional neural organization design, to contrast our strategy with existing streamlining strategies. AlexNet is comprised of five convolutional layers and three totally associated layers. More detail on the engineering might be found in the article [2]. Since Alex Net is a provoking profound neural organization to assemble, we need to expand our way to deal with this current organization's plan. Figure 2 shows the aftereffects of applying our technique over SGD. We review that our framework accomplishes a lot higher precision and diminished misfortune after 100,000 and 200,000 cycles.

International Journal of application or Innovation in engineering and management

Ijaiem.com/Aug 2024/ Volume 13/Issue 2/Article No-1/35-41

ISSN: 2319-4847

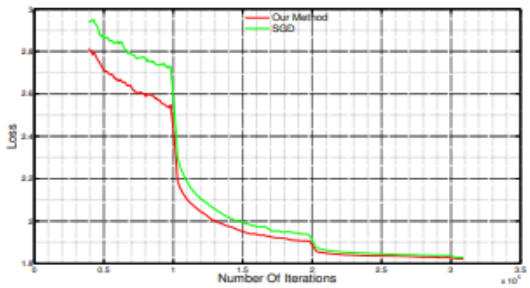
Conversely, we are as yet ready to accomplish the greatest precision of 57.5 percent on the approval set after 295,000 cycles, while SGD just finishes after 345,000 emphasess, yielding in a 15% reduction in preparing time. Given that a major model takes over seven days to completely prepare, this is a huge investment funds. Our misfortune is reliably lower than SGD all through all emphasess. For each 100,000 emphasess in the current model, we do a stage somewhere around a factor of ten. We change the quantity of preparing emphasess at a specific learning speed till we lead a stage down to evaluate how our methodology proceeds as we decline the quantity of preparing cycles. Table III shows a definitive precision after 350,000 cycles of SGD and our methodology. In any case, when the quantity of cycles is diminished and the learning speed is eased back, the last exactness falls to some degree, demonstrating that our strategy produces more prominent precision than SGD. Note that we just report precision to the best 1 class. Since we use the Cafe execution of the Alex Net structure and don't utilize any information expansion techniques, our outcomes are to some degree lower than those detailed in [2].

CONCLUSION

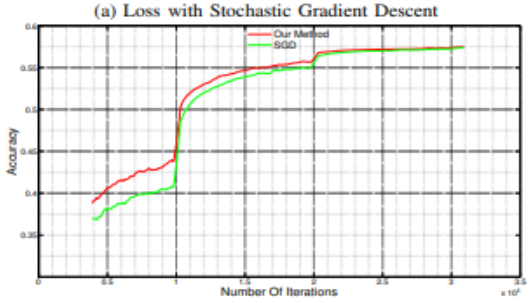
This paper proposes a nonexclusive strategy for preparing profound neural organizations that utilizations layer-explicit versatile learning rates.

Iteration	SGD	Ours-SGD	Nesterov	Ours-NAG	AdaGrad	Ours-AdaGrad
5000	68.8 ± 0.49	70.10 ± 0.89	69.36 ± 0.31	70.10 ± 0.59	54.90 ± 0.26	57.53 ± 0.67
10000	74.05 ± 0.51	74.48 ± 0.59	73.17 ± 0.25	74.00 ± 0.29	58.26 ± 0.58	60.95 ± 0.59
25000	77.40 ± 0.32	77.43 ± 0.15	76.17 ± 0.61	77.29 ± 0.59	63.02 ± 0.95	64.90 ± 0.57
60000	78.76 ± 0.87	78.74 ± 0.38	78.35 ± 0.33	78.18 ± 0.65	66.86 ± 0.93	68.03 ± 0.23
70000	81.78 ± 0.14	82.10 ± 0.32	81.75 ± 0.25	81.92 ± 0.26	67.04 ± 0.91	68.30 ± 0.39

TABLE II: Mean accuracy on CIFAR10 as seen in the table after multiple iterations for SGD, NAG and AdaGrad with layer-specific adaptive models. There is a report of the mean and standard deviation over 5 runs.



(a) Loss with Stochastic Gradient Descent



(b) Accuracy with Stochastic Gradient Descent

Fig. Fig. 2: Data collection on Image Net: plot relating stochastic gradient descent to our layer-wise adaptive learning speeds. Throughout all iterations, we can see a clear increase in precision and loss over the standard SGD process.

Iterations	SGD	Our Method
70,000	55.72%	55.84%
80,000	56.25%	56.57%
90,000	56.96%	57.13%

TABLE III: Contrast of stochastic inclination plunge and our progression down approach at different cycles on Image Net, which can be utilized with a worldwide learning rate on top of any advancement strategy.

To figure a versatile learning rate for each layer, the framework utilizes slopes from each layer. At the point when the boundaries are in a low ebb and flow saddle point region, it plans to accelerate assembly. Layer-explicit learning rates regularly empower the framework to abstain from slow learning, commonly actuated by tiny inclination esteems, in the underlying layers of the profound organization.

REFERENCES

[1] R. Pascanu, Y. N. Dauphin, S. Ganguli, and Y. Bengio, "On the saddle point problem for non-convex optimization," arXiv preprint arXiv:1405.4604, 2014.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105.

[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. IEEE, 2014, pp. 1701–1708.

[4] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Citeseer, 2013, pp. 1631–1642.

[5] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," Signal Processing Magazine, IEEE, vol. 29, no. 6, pp. 82–97, 2012.

[6] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier networks," in Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. JMLR W&CP Volume, vol. 15, 2011, pp. 315–323.

[7] H. Robbins, S. Monro et al., "A stochastic approximation method," The Annals of Mathematical Statistics, vol. 22, no. 3, pp. 400–407, 1951.

[8] M. D. Zeiler, "Adadelta: An adaptive learning rate method," arXiv preprint arXiv:1212.5701, 2012.

[9] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," The Journal of Machine Learning Research, vol. 12, pp. 2121–2159, 2011.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] A. J. Bray and D. S. Dean, "Statistics of critical points of gaussian fields on large-dimensional spaces," Physical review letters, vol. 98, no. 15, p. 150201, 2007.

[12] Y. V. Fyodorov and I. Williams, "Replica symmetry breaking condition exposed by random matrix calculation of landscape complexity," Journal of Statistical Physics, vol. 129, no. 5-6, pp. 1081–1116, 2007. [13] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," USSR Computational Mathematics and Mathematical Physics, vol. 4, no. 5, pp. 1–17, 1964.

[14] Y. Nesterov, "A method of solving a convex programming problem with convergence rate o (1/k2)," in Soviet Mathematics Doklady, vol. 27, no. 2, 1983, pp. 372–376.

[15] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in Proceedings of the 30th International Conference on Machine Learning (ICML-13), 2013, pp. 1139–1147.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," arXiv preprint arXiv:1408 .5093, 2014.