Исследование изменения пространства параметров при дообучении и модификации моделей глубокого обучения*

***Иванов И.С., Бахтеев О.Ю.*[1]*, Стрижов В.В.*[2]**

Статья посвящена задаче выбора оптимальной модели классификации временных рядов с использованием глубокого обучения. Рассматривается суперпозиция моделей, относящихся к следующим классам нейронных сетей: автокодировщики и двухслойные нейронные сети. Предлагается исследовать дисперсию и матрицу ковариаций параметров модели при различных подходах к поэтапному обучению, на основе чего предложить эффективный способ модификации модели глубокого обучения.

Ключевые слова: *выбор модели классификации; пространство параметров; дисперсия; ковариационная матрица; нейронные сети глубокого обучения.*

1. Введение

В данной работе рассматривается задача выбора и оптимизации модели классификации временных рядов. Под временным рядом понимается упорядоченный набор измерений некоторой величины, в котором каждое измерение соответствует определенному моменту времени. Исходная модель описана в статье [4]. Модель использует нейронные сети глубокого обучения для выделения информативных признаков и классификации временных рядов. Она представляет собой многоуровневую суперпозицию автокодировщиков [7] и двухслойных нейронных сетей [14]. Все уровни суперпозиции, кроме последнего, обучаются по принципу «обучение без учителя» и участвуют в построении признакового пространства. Последний уровень суперпозиции обучается «с учителем» и решает задачу классификации по признакам, выделенным на нижних уровнях суперпозиции.

Предобучением называется предварительное обучение модели по принципу «обучение без учителя». Предобучение [6] слоев модели помогает избавиться от шумов во временных рядах и выделить необходимые признаки, тем самым стабилизируя параметры модели для дальнейшего дообучения (обучения всей сети по принципу «обучение с учителем»). Для поиска оптимальной модификации модели исследуется дисперсия и матрица ковариаций параметров нейросети на разных уровнях при различных подходах к поэтапному обучению. Строится add-del метод (метод добавления/удаления признаков) [8] по полученным данным. Проблемой данного анализа является большой размер ковариационных матриц.

Проблема выбора оптимальной модели является достаточно изученной [16, 17, 18]. Существуют различные методы оптимизации нейронной сети, в том числе такие техники прореживания сети, как Optimal Brain Damage [5] и Optimal Brain Surgeon [15]. Для оптимизации гиперпараметров используют алгоритмы случайного поиска и некоторые жадные алгоритмы [9, 10]. При рассмотрении нейронных сетей как вероятностных моделей [11], применяются байевские методы оптимизации [12].

В данной работе поставлен вычислительный эксперимент на двух выборках – временных рядах акселерометра и временных рядах акселерометра и гироскопа мобильного телефона.

---

[1]консультант

[2]научный руководитель и постановщик задачи

2. Постановка задачи

Дана выборка $\mathfrak{D} = \{(\mathbf{x}_i, \mathbf{t}_i)\}, i \in \mathcal{I} = \{1 \dots m\}$, состоящая из m объектов, каждый из которых описывается n признаками $\mathbf{x}_i \in \mathbb{R}^n$ и принадлежит одному из z классов $\mathbf{t}_i \in \{0,1\}^z$. Также задано разбиение множество индексов выборки $\mathcal{I} = \mathcal{L} \sqcup \mathcal{T}$ на обучающую $(\mathbf{x}_i, \mathbf{t}_i), i \in \mathcal{L}$ и контрольную $(\mathbf{x}_i, \mathbf{t}_i), i \in \mathcal{T}$. Требуется выбрать устойчивую модель классификации оптимальной сложности.

Определение 1. *Моделью назовем отображение:*

$$\mathbf{f} : (\mathbf{w}, \mathbf{X}) \mapsto \mathbf{y},$$

где $\mathbf{w} = [w_1, \dots, w_j, \dots, w_n]^\intercal, j \in \mathcal{J} = \{1, \dots, n\}$ *— вектор параметров модели,* $\mathbf{X} \in \mathbb{R}^{n \times m}$ *— матрица плана,* $\mathbf{y} \in \{0,1\}^z$ *— зависимая переменная.*

В данной работе рассматриваются модели, представляющие собой многоуровневую суперпозицию автокодировщиков и двухслойных нейронных сетей с функциями активации tanh и softmax:

$$\mathbf{a}(\mathbf{x}) = \mathbf{W}_2^\intercal \mathbf{tanh}(\mathbf{W}_1^\intercal \mathbf{x}),$$

$$\mathbf{p}(\mathbf{x}) = \frac{\exp(\mathbf{a}(\mathbf{x}))}{\sum_j \exp(a_j(\mathbf{x}))}.$$

Вектор $\mathbf{p}$ интерпретируется как вектор вероятностей: p_ξ есть вероятность того, что вектор $\mathbf{x}$ принадлежит классу с номером ξ:

$$\mathbf{p}(\mathbf{x}) = \{p_\xi\}, \quad 0 \leqslant p_\xi \leqslant 1, \quad \sum p_\xi = 1, \quad \xi = 1 \dots z.$$

Под вектором параметров модели будем понимать $\mathbf{w} = \mathrm{vec}(\mathbf{W}_1^\intercal | \dots | \mathbf{W}_2^\intercal)$, где $\mathbf{W}_i$ — матрица весов i-го слоя нейронной сети. Вектор $\mathbf{y} = [y_1, \dots, y_\xi, \dots, y_z]^\intercal$ определим следующим образом:

$$y_\xi = \begin{cases} 1, & \text{если } \xi = \underset{\xi \in \{1,\dots,z\}}{\operatorname{argmax}}(p_\xi), \\ 0, & \text{иначе.} \end{cases}$$

Под структурными параметрами модели будем понимать количество нейронов во внутренних слоях автокодировщиков и двухслойных нейронных сетей — N_i.

Определение 2. *Параметр* w_j *модели* $\mathbf{f}$ *назовем активным, если* $w_j \neq 0$.

Определение 3. *Структурой* $\mathcal{A}$ *модели* $\mathbf{f}$ *назовем множество индексов активных параметров этой модели* $\mathcal{A} = \{j : w_j \neq 0\} \subseteq \mathcal{J}$.

Каждая структура $\mathcal{A} \subseteq \mathcal{J}$ однозначно задает некоторую модель:

$$\mathbf{f}_\mathcal{A} : \hat{\mathbf{w}}_\mathcal{A} \in \mathbb{R}^n,$$

где $\mathbf{f}_\mathcal{A}$ — модель со структурой $\mathcal{A}$, а $\hat{\mathbf{w}}_\mathcal{A} \in \mathbb{R}^n$ — оптимальный вектор параметров модели $\mathbf{f}_\mathcal{A}$, определение которому будет дано ниже. Объединение всех $\mathbf{f}_\mathcal{A}$ назовем множеством допустимых моделей:

$$\mathfrak{F} = \bigcup_{\mathcal{A} \subseteq \mathcal{J}} \{\mathbf{f}_\mathcal{A}\}. \tag{1}$$

Оптимальную модель $\hat{\mathbf{f}}_\mathcal{A}$ будем выбирать из множества допустимых моделей $\mathfrak{F}$.

В качестве функции ошибки выберем функцию:

$$S(\mathbf{w}|\mathcal{K}) = -\sum_{i\in\mathcal{K}}\sum_{\xi=1}^{z} t_{i\xi}\ln\bigl(p_\xi(\mathbf{x}_i,\mathbf{w})\bigr), \tag{2}$$

максимизирующую логарифм правдоподобия случайной величины $\mathbf{y}$ и заданную на разбиении выборки $\mathfrak{D}$, определенном некоторым множеством индексов $\mathcal{K} \subseteq \mathcal{I}$, $\mathbf{t}_i = [t_{i1},\ldots,t_{i\xi},\ldots,t_{iz}]^\intercal$.

Определение 4. *Оптимальным вектором параметров модели $\mathbf{f}_\mathcal{A}$ назовем такой вектор $\hat{\mathbf{w}}_\mathcal{A}$, который является решением следующей задачи оптимизации:*

$$\hat{\mathbf{w}}_\mathcal{A} = \underset{\mathbf{w}_\mathcal{A}\in\mathbb{R}^k}{\operatorname{argmin}} S(\mathbf{w}_\mathcal{A}|\mathcal{L}). \tag{3}$$

Для оценки качества моделей и сравнения их друг с другом введём два критерия качества — устойчивость и точность.

Определение 5. *Устойчивостью $\eta = \eta(\hat{\mathbf{w}})$ модели $\mathbf{f}$ с вектором параметров $\mathbf{w}$ назовем число η, равное числу обусловленности матрицы ковариаций $\boldsymbol{Cov}(\mathbf{w})$, т.е. $\eta(\hat{\mathbf{w}}) = \frac{\lambda_{\max}}{\lambda_{\min}}$, где $\lambda_{\max}$ — максимальное, а $\lambda_{\min}$ — минимальное собственное число матрицы $\boldsymbol{Cov}(\mathbf{w})$.*

Чем лучше обусловлена матрица $\mathbf{Cov}(\mathbf{w})$, тем более устойчива модель. У идеально устойчивой модели $\lambda_{\min} = \lambda_{\max},\ \eta = 1$.

Определение 6. *Под точностью S модели $\mathbf{f}$ с вектором параметров $\hat{\mathbf{w}}$ будем понимать величину, обратную функции ошибки (2) на контрольной выборке.*

Чем больше значение функции ошибки, тем меньше точность модели. Задача выбора оптимальной модели состоит в том, чтобы найти модель, имеющую максимальную точность, из множества допустимых моделей $\mathfrak{F}$.

3. Стратегия жадной модификации модели

Наша стратегия модификация модели задается следующими математическими объектами:

— критериями устойчивости η и точности S,
— набором ограничений на структуру и параметры модели $\mathcal{A} \subseteq \mathcal{J}$, $\mathbf{w} = \hat{\mathbf{w}}_\mathcal{A}$ из (3),
— критерием устойчивого прореживания (5).

Действуя согласно стратегии, мы будем изменять структуру модели, удаляя из неё элементы. Для определения индекса параметра $\hat{j}$, который должен быть удален из модели, предлагается следующий критерий оптимизации модели.

3.1 Критерий устойчивого прореживания

Критерий устойчивого прореживания основан на модификации метода Белсли и оценке ковариационной матрицы параметров.

Пусть $\mathbf{W}$ — матрица реализаций оптимального вектора параметров $\hat{\mathbf{w}}$, определенного в (3). Пусть эта матрица имеет размерность $r \times k$. По реализациям вектора параметров вычисляется выборочная матрица ковариаций - $\mathbf{Cov}(\mathbf{w})$. Предполагается, что она блочно-диагональный вид, где блоки образованы нейронами (входящими в них весовыми функциями). При этом близость элемента матрицы к нулю трактуется как независимость соответствующих параметров.

Оценками дисперсии параметров будут диагональные элементы $\mathbf{Cov}(\mathbf{w})$:

$$\sigma(w_\zeta) = \mathbf{Cov}(\mathbf{w})_{\zeta\zeta}.$$

Выполним сингулярное разложение $\mathbf{Cov}(\mathbf{w})$:

$$\mathbf{Cov}(\mathbf{w}) = \mathbf{U}\mathbf{S}\mathbf{V}^\intercal, \tag{4}$$

где $\mathbf{U}$ и $\mathbf{V}$ — ортогональные матрицы размера $r \times r$ и $k \times k$ соответственно, а $\mathbf{\Lambda}$ — матрица, на диагонали которой стоят сингулярные числа матрицы $\mathrm{cov}(\mathbf{w})$.

Индексом обусловленности η_ζ назовём отношение максимального элемента $\lambda_{\max}$ матрицы $\mathbf{\Lambda}$ к ζ-ому по величине элементу λ_ζ этой матрицы:

$$\eta_\zeta = \frac{\lambda_{\max}}{\lambda_\zeta}.$$

Долевой коэффицент $q_{\zeta j}$ определим как вклад j-го признака в дисперсию ζ-го элемента вектора параметров $\mathbf{w}$:

$$q_{\zeta j} = \frac{u_{\zeta j}^2 \lambda_{jj}^2}{\sigma(w_\zeta)}.$$

Найдём индексы обусловленности и долевые коэффициенты для набора активных параметров $\mathcal{A}$. Большие значения индексов обусловленности указывают на зависимость между признаками. Поэтому для нахождения параметра, отвечающего этому критерию прореживания, находим максимальный индекс обусловленности:

$$\hat{\zeta} = \operatorname*{argmax}_{\zeta \in \mathcal{A}} \eta_\zeta.$$

Затем находим максимальный долевой коэффициент, соответствующий найденному максимальному индексу обусловленности $\eta_{\hat{\zeta}}$:

$$\hat{j} = \operatorname*{argmax}_{j \in \mathcal{A}} q_{\hat{\zeta} j}, \tag{5}$$

Параметр $w_{\hat{j}}$ и есть параметр, отвечющий критерию устойчивого прореживания.

3.2 Описание базовой стратегии

Стратегия жадной модификации модели состоит из последовательности этапов Del.

Этап Del. Во множестве активных параметров ищем параметр с индексом $\hat{j}$, отвечающий критерию прореживания (5) и удаляем его из множества активных параметров:

$$\mathcal{A} = \mathcal{A} \backslash \hat{j}.$$

Сперва модель предобучается, приобретая вектор параметров $\mathbf{w}_0$. После этого она независимо дообучается l раз, приобретая вектора параметров $\mathbf{w}_1, ..., \mathbf{w}_l$, на основе которых мы и вычисляем выборочную ковариационную матрицу. Далее повторяем этап Del некоторое заданное число раз, исключая параметры из вектора $\mathbf{w}_0$. После чего остается дообучить модель с модифицированным пространством параметров и измерить её качество.

4. Вычислительный эксперимент

В вычислительном эксперименте использовался набор временных рядов с акселерометра мобильного устройства – WISDM [20]. Цель вычислительного эксперимента заключалась в проверке эффективности предложенной стратегии модификации модели и сравнения полученных результатов с другими методами.

4.1 Программное обеспечение

Эксперимент производился на ПО [21], предназначенном для создания, обучения и оценки моделей глубокого обучения.

4.2 Эксперимент

Набор состоял из сегментов временных рядов акселерометра мобильного телефона, каждый из которых описывал один из четырех типов физической активности человека – ходьбу, бег, стояние и сидение. Сегмент представлял из себя 10 секундный отрезок исходного временного ряда и состоял из 600 измерений – по 200 измерений проекции ускорения на каждую из координатных осей. С более подробным описанием данных можно ознакомиться в работе [20].

Сбалансированная выборка состояла из 12546 объектов. В эксперименте использовалась трёхслойная суперпозиция, состоящая из двух автокодировщиков и однослойной нейросети с нелинейной функцией активации с 200, 6 и 6 нейронами в каждом слое соответственно. Автокодировщики предобучались на 1000 объектах. Дообучение реализовывалось в рамках кросс-валидации на оставшихся 11546 объектах, разделение на обучающую и контрольную выборку проводилось случайным образом в соотношении 3:1. Было произведено 15 итераций кросс-валидации, на основе изменений вектора параметров были посчитаны дисперсии компонент вектора, гистограммы для параметров первых двух слоёв изображены на рис. 1.

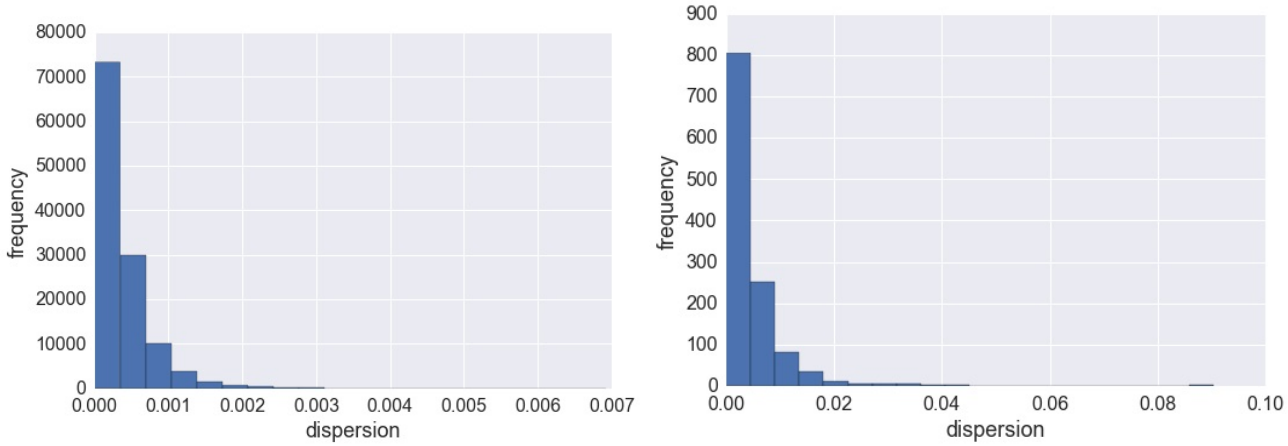

Рис. 1. Распределение дисперсий весовых функций по слоям

Далее измерялось влияние удаления (обнуления) части параметров сети на качество работы модели. В эксперименте удалялось от 10 до 70 процентов всех параметров. Для сравнения качества работы модели измерялась точность модели при классификации тестовой выборки. Предложенным методом выбора параметров на удаление является модифицированный метод Белсли, описанный в разделе 3.1. Для сравнения также были проведены эксперименты с другими методами. Из предположения, что неустойчивые (имеющие

большую дисперсию) параметры могут оказывать наименьшее влияние на модель, в одном из экспериментов удалялись параметры с наибольшей дисперсией. В остальных экспериментах к удалению были отобраны параметры с наименьшей дисперсией и случайно выбранные параметры. Результаты измерений изображены на рис. 2.

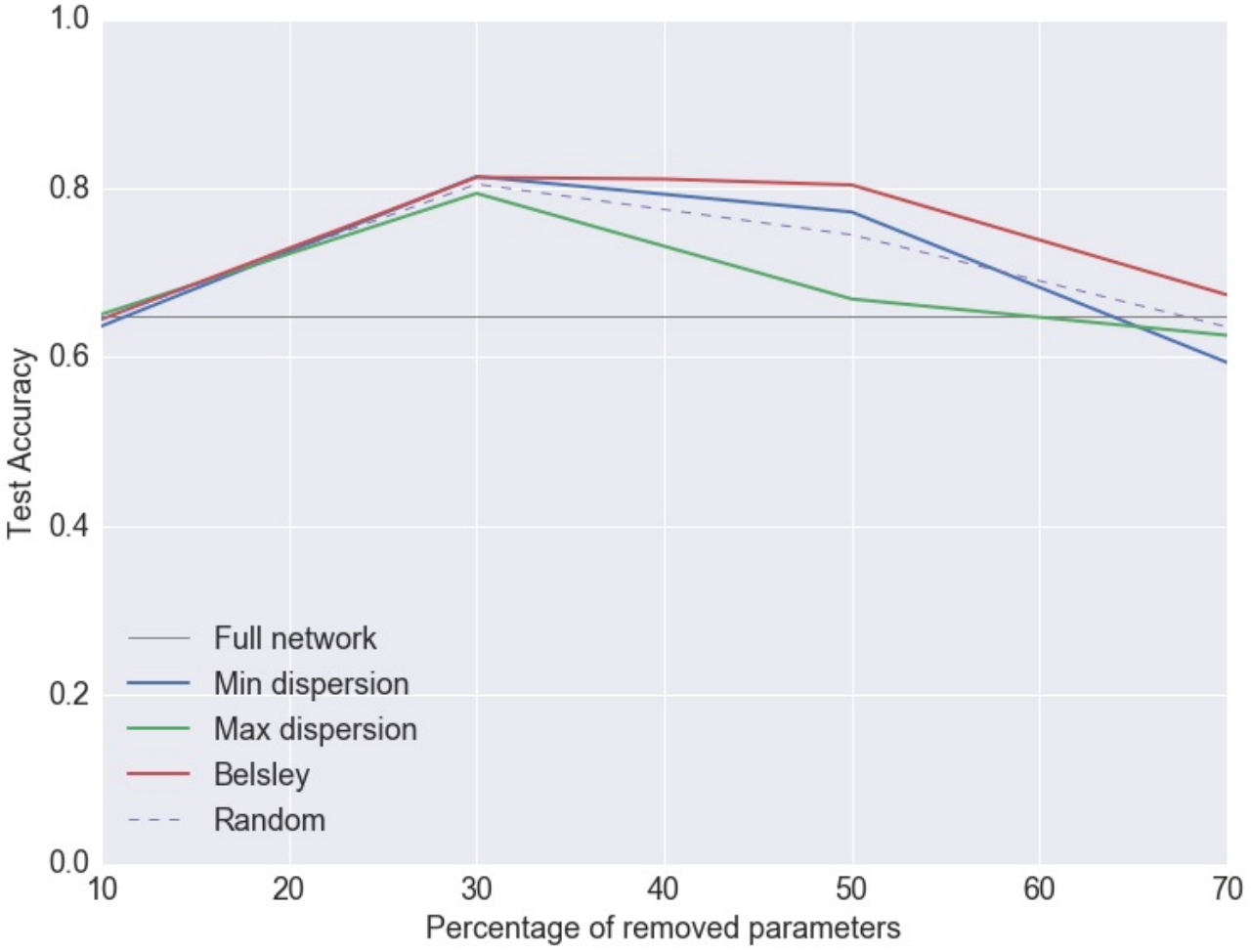

Рис. 2. Зависимость точности модели от процента отброшенных параметров

Из графика видно, что при помощи модификации модели удалось увеличить её точность. При этом предложенный метод показал лучшие результаты.

4. Заключение

В данной работе рассмотрена задача классификации временных рядов при помощи глубокой нейронной сети. Исследовано влияние различных методов разряжения пространства параметров на качество модели глубокого обучения. Предлагаемый метод, основанный на методе Белсли, показал относительно хороший результат и, имея теоретические обоснования, может быть использован в других задачах.

В дальнейшем планируется проанализировать работу метода при пошаговой модификации.

Литература

[1] *Задаянчук А. И., Попова М. С., Стрижов В. В.* Выбор оптимальной модели классификации физической активности по измерениям акселерометра http://strijov.com/papers/Zadayanchuk2015OptimalNN4.pdf

[2] *Попова М. С., Стрижов В. В.* Построение сетей глубокого обучения для классификации временных рядов http://strijov.com/papers/PopovaStrijov2015DeepLearning.pdf

[3] *Бахтеев О. Ю., Попова М. С., Стрижов В. В.* Системы и средства глубокого обучения в задачах классификации http://svn.code.sf.net/p/mlalgorithms/code/Group074/Bakhteev2015TheanoCuda/doc/Bakhteev2015TheanoAWS.pdf

[4] *Попова М. С., Стрижов В. В.* Выбор оптимальной модели классификации физической активности по измерениям акселерометра http://svn.code.sf.net/p/mlalgorithms/code/Group174/Popova2014OptimalModelSelection/doc/Popova2014OptimalModelSelection.pdf

[5] *LeCun Y.* Optimal Brain Damage http://yann.lecun.com/exdb/publis/pdf/lecun-90b.pdf

[6] *Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, Samy Bengio* Why Does Unsupervised Pre-training Help Deep Learning? *Journal of Machine Learning Research 11 (2010) 625-660* http://jmlr.org/papers/volume11/erhan10a/erhan10a.pdf

[7] *Andrew Ng* Sparse autoencoder *CS294A Lecture notes* https://web.stanford.edu/class/cs294a/sparseAutoencoder.pdf

[8] *Воронцов К.В.* Оценивание моделей и отбор признаков http://www.machinelearning.ru/wiki/images/archive/4/4f/20111004204412!Voron-ML-Modeling-slides.pdf

[9] *James Bergstra, Remi Bardenet, Yoshua Bengio, Balazs Kegl* Algorithms for Hyper-Parameter Optimization http://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.pdf

[10] *James Bergstra, Yoshua Bengio* Random Search for Hyper-Parameter Optimization *Journal of Machine Learning Research 13 (2012) 281-305* http://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf

[11] *David J.C. MacKay* Bayesian Methods for Neural Networks: Theory and Applications http://www.inference.eng.cam.ac.uk/mackay/cpi4.pdf

[12] *Yarin Gal, Zoubin Ghahramani* Dropout as a Bayesian Approximation: Insights and Applications http://mlg.eng.cam.ac.uk/yarin/PDFs/Dropout_as_a_Bayesian_approximation.pdf

[13] *Naftali Tishby, Noga Zaslavsky* "Optimal Deep Learning" and the Information Bottleneck method *ICRI-CI retreat, Haifa, May 2015* http://icri-ci.technion.ac.il/files/2015/05/01-Tali-Tishby-1505051.pdf

[14] *G. Brightwell, C. Kenyon, H. Paugam-Moisy* Multilayer neural networks: one or two hidden layers? http://papers.nips.cc/paper/1239-multilayer-neural-networks-one-or-two-hidden-layers.pdf

[15] *Babak Hassibi, David G. Stork, Gregory J Wolff* Optimal Brain Surgeon and General Network Pruning http://ee.caltech.edu/Babak/pubs/conferences/00298572.pdf

[16] *Quoc V. Le, Jiquan Ngiam, Adam Coates, Abhik Lahiri, Bobby Prochnow, Andrew Y. Ng* On Optimization Methods for Deep Learning http://machinelearning.wustl.edu/mlpapers/paper_files/ICML2011Le_210.pdf

[17] *M. Bashiri* Tuning the parameters of an artificial neural network using central composite design and genetic algorithm http://www.sciencedirect.com/science/article/pii/S1026309811002136

[18] *Misha Denil, Babak Shakibi, Laurent Dinh, Marc'Aurelio Ranzato, Nando de Freitas* Predicting Parameters in Deep Learning `http://www.cs.toronto.edu/~ranzato/publications/denil_nips2013.pdf`

[19] *Anguita D., Ghio A., Oneto L., Parra X., Luis Reyes-Ortiz J.* Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine *Ambient Assisted Living and Home Care: Proceedings of the 4th International Workshop (IWAAL 2012). – Springer, 2012. Vol. 7657. P. 216–223*

[20] *Kwapisz J. R., Weiss G. M., Moore S.* Activity recognition using cell phone accelerometers *SIGKDD Explorations, 2010. Vol. 12. No 2. P. 74–82*

[21] Python-driven software for time-series classification `http://svn.code.sf.net/p/mlalgorithms/code/Group074/Bakhteev2015Sysdocs`