

ОСОБЕННОСТИ ИНИЦИАЛИЗАЦИИ СВЕРТОЧНЫХ СЛОЕВ С ФУНКЦИЕЙ АКТИВАЦИИ RELU В НЕЙРОСЕТЕВЫХ МОДЕЛЯХ ДЛЯ РАСПОЗНАВАНИЯ ШУМОВЫХ СИГНАЛОВ

Р.Ю. Порцев

Институт проблем управления им. В.А. Трапезникова РАН
Россия, 117997, Москва, Профсоюзная ул., 65
E-mail: poruss@ipu.ru

А.В. Макаренко

Институт проблем управления им. В.А. Трапезникова РАН
Россия, 117997, Москва, Профсоюзная ул., 65
E-mail: avm.science@mail.ru

Ключевые слова: сверточные нейросети, Relu, инициализация весов, стохастические сигналы.

Аннотация: В работе рассматривается проблема обучаемости многослойных сверточных нейросетей с функцией активации Relu в задаче распознавания шумовых сигналов. При малом количестве нейронов в сверточных слоях обучаемость сети существенно зависит от ее инициализации, особенно эта зависимость проявляется в глубоких сверточных сетях. Предложены новые стратегии инициализации весов в сверточных слоях, позволяющие повысить обучаемость сетей. Проведен сравнительный анализ стратегий инициализации весов на основе характеристик обученных нейросетей.

1. Введение

В настоящее время, наряду с «тяжелыми» сложными нейросетевыми моделями, всё существеннее возникает потребность в «легких» моделях для решения различных проблем в условиях реального времени и/или ограниченных аппаратных средств [1-3], включая маломощные и мобильные сенсорные, робототехнические комплексы. При этом ряд задач оперируют не детерминированными сигналами на фоне случайных помех, а непосредственно стохастическими сигналами в условиях случайных помех, например, в задачах мониторинга городской обстановки при распознавании шума окружающей среды [4], в задачах управления технологическим процессом при обнаружении и диагностике неисправностей [5-7] и т.п.

Ранее в работе [8] авторами было показано, как элементарные сверточные нейросети успешно решают задачу классификации шумовых сигналов трех типов, а в работе [9] авторами изучались многослойные сверточные нейросети, в которых была обозначена проблема плохой обучаемости многослойных сверточных сетей с функцией активации Relu. Настоящая работа посвящена дальнейшему исследованию этой проблемы.

2. Описание задачи

2.1. Набор данных

Введём в рассмотрение псевдослучайные последовательности трёх типов вероятностных распределений: \mathcal{N} – нормального, \mathcal{U} – равномерного, \mathcal{E} – экспоненциального. Выбор данных типов распределений обусловлен рядом соображений, представленных в [8].

Набор данных генерировался функциями из библиотеки NumPy, после чего каждая из последовательностей в наборе данных стандартизировалась. Полное множество разбивалось на неперекрывающиеся выборки: обучающая – 75 000 сэмплов на класс, тестовая – 25 000 сэмплов на класс. В обеих выборках распределение классов равновероятное. Длина всех последовательностей была принята в 1 024 отсчёта.

С целью оценивания устойчивости нейросетей, проведем эксперимент по их функционированию на загрязнённых данных. При этом эксперимент будем проводить на предварительно обученных сетях (на чистых данных). Стоит отметить, что выбор сетей, обученных на чистых данных, для целей эксперимента предназначен для имитации условий априорной неопределенности при принятии решений. В качестве модели загрязнения примем наличие импульсных помех в канале связи и/или на сенсоре, связанных с запирающим каналом и/или сенсора. Формально данная модель описывается выражением:

$$(1) \quad x_{dk} = \begin{cases} 0 & k \in K', \alpha' = \frac{\|K'\|}{K}, k = \overline{0, K-1}, \\ x_k & \text{иначе,} \end{cases}$$

где K' – некоторое количество отсчётов в последовательности в псевдослучайном порядке (без повторов), которое заменяется нулями; α' – уровень загрязнения данных.

2.2. Архитектуры нейросетей

При проведении исследования рассматривались 1) двухслойные и 2) трехслойные сверточные нейросети прямого распространения с полносвязным выходным слоем с 3-мя нейронами (по числу распознаваемых классов сигналов). В первом случае количество ядер в сверточных слоях варьировалось от 1 до 2, таким образом, были рассмотрены архитектуры $M=\{11, 12, 21, 22\}$. Во втором случае количество ядер варьировалось от 1 до 3, при этом были рассмотрены архитектуры $M=\{111, 123, 321, 333\}$. Во всех сверточных слоях применялась функция активации Relu, в выходном полносвязном слое – SoftMax [10]. Для обучения использовался подход одноклассовой классификации метки и минимизировалась функция потерь L в виде категориальной кросс-энтропии [10].

2.3. Стратегии инициализации сверточных слоев

Особенность использования функции активации Relu в многослойных сверточных нейросетях заключается в том, что на выходе n -го сверточного слоя с функцией активации Relu формируется карта признаков с неотрицательными значениями:

$$v_i(x, w_i, b_i) = \max(0, w_i x + b_i), v_i \geq 0,$$

где w_i – веса (kernels) и b_i – смещения (bias) сверточного ядра.

В случае положительных входных данных $x_i < 0$, отрицательных значений весов $w_{i+1} < 0$, и нулевых смещений $b_{i+1} = 0$ – последующего слоя карта признаков на выходе такого слоя «обнуляется», что приводит к «обнулению» градиентов при обучении нейросети. Другими словами обучение нейросети «останавливается». Описанная ситуация может возникнуть при стандартной инициализации [11] сверточных слоев, особенно в нейросетях с малым числом и размером сверток, в частности данная проблема была обозначена в работе [9].

Для решения данной проблемы предлагается 2 варианта инициализации сверточных слоев: 1) положительными значениями смещений сверточных слоев начиная со второго $b_{i+1} > 0$; 2) положительными значениями весов сверточных слоев начиная со второго $w_{i+1} > 0$.

Таким образом, в работе сравнивались три стратегии инициализации сетей:

- А** инициализация по умолчанию [11]: веса (kernels) задавались с помощью равномерного распределения (симметричного), а смещения (bias) – нулями: $w_i = \text{glorot_uniform}$, $b_i = 0$;
- Б** инициализация положительными значениями смещений (bias) второго и последующего сверточных слоев $w_{i+1} = \text{glorot_uniform}$, $b_{i+1} > 0$;
- В** инициализация положительными значениями весов (kernels) второго и последующего сверточных слоев: $w_{i+1} > 0$, $b_{i+1} = 0$.

3. Результаты

3.1. Обучение нейросетей

Обучение нейросетей проводилось на Keras Framework [12] с Google TensorFlow [13] версии 2.14.1. Каждая архитектура инициализировалась пятью разными вариантами весов, задаваемых через параметр seed генератора псевдослучайных чисел. Каждый из инициализированных вариантов обучался семью оптимизаторами (на стандартных настройках фреймворка Keras) в течение 100 эпох каждый, в сумме 35-ти попыток обучения. После обучения нейросетей рассчитывались их характеристики: значения метрики F_1 и функции потерь L по объединенному тренировочному и тестовому набору.

На рис. 1 показано распределение «плохих» сетей с $F_1 = 0$ (рис. 1а) и «хороших» сетей с $F_1 = 1$ (рис. 1б) в зависимости от архитектуры и стратегии инициализации сверточных слоев. На рис. 1а хорошо видно, что применение предложенных стратегий: **Б** или **В** – для многослойных сверточных нейросетей положительно сказывается на их обучаемости по сравнению со стратегией по умолчанию. Стратегия **В** в целом показывает лучшие результаты по сравнению со стратегией **Б**: наименьшее количество не обученных сетей и наибольшее количество обученных сетей с качеством $F_1 = 1$. Стоит заметить, что количество «хороших» сетей больше зависит не от стратегии инициализации, а от архитектуры сети, при этом большую роль играет размер последнего сверточного слоя: «чем больше ядер, тем лучше», причем это относится ко всем стратегиям инициализации сверточных слоев.

F1=0	А	Б	В
M=11	21	1	2
M=12	10	3	1
M=21	10	3	0
M=22	5	0	1
M=111	29	4	1
M=123	15	0	2
M=321	16	11	2
M=333	8	0	1

(а)

F1=1	А	Б	В
M=11	1	3	1
M=12	6	9	22
M=21	3	2	1
M=22	10	15	20
M=111	1	0	2
M=123	8	15	22
M=321	1	1	1
M=333	17	23	25

(б)

Рис. 1. Количество нейросетей со значением метрики $F_1 = 0$ (а) и $F_1 = 1$ (б) в зависимости от архитектуры и стратегии инициализации сверточных слоев.

3.2. Top-2 нейросети

Далее, для каждого варианта архитектуры и стратегии инициализации отбирались 2 наилучшие сети (top-2) по правилу $\{\max F_1, \min L\}$ по объединенному тренировочному и тестовому набору, после чего сравнение характеристик обученных нейросетей из top-2 выполнялось уже по тестовой выборке.

Практически все сети top1 имеют максимальное значение метрики $F_1 = 1$, за исключением архитектуры $M = 111$ в случае стратегии **Б**, при этом наименьшее значение $F_1 \approx 0,999893$ также имеет данный вариант архитектуры и стратегии инициализации, но уже сеть top2. Лучший показатель $L \approx 2,953 \cdot 10^{-9}$ ожидаемо имеет сеть с архитектурой $M = 333$, но при стратегии **Б**, в которой смещения сверточных слоев инициализировались положительными значениями. Стоит заметить, что сильнее всего выделяется зависимость L от архитектуры, при этом большую роль играет размер последнего сверточного слоя, как и в случае с количеством «хороших» сетей.

А				Б				В			
	F1	Loss	opt		F1	Loss	opt		F1	Loss	opt
M=11	1	9,318E-03	sgd	M=11	1	1,352E-06	adamax	M=11	1	9,516E-03	sgd
	0,999987	5,490E-05	rmsprop		1	9,135E-03	sgd		0,999933	4,739E-04	nadam
M=12	1	1,956E-07	adam	M=12	1	2,092E-06	adamax	M=12	1	7,176E-08	adam
	1	2,418E-07	nadam		1	2,689E-06	adamax		1	1,241E-07	adam
M=21	1	3,657E-07	nadam	M=21	1	1,286E-06	nadam	M=21	1	9,502E-03	sgd
	1	2,326E-06	adamax		1	1,365E-06	adamax		0,999973	2,622E-03	nadam
M=22	1	3,313E-07	adamax	M=22	1	3,284E-08	nadam	M=22	1	4,290E-08	adam
	1	5,529E-07	adamax		1	3,371E-08	adam		1	9,335E-08	adam
M=111	1	9,037E-07	adamax	M=111	0,999973	6,003E-05	rmsprop	M=111	1	9,345E-06	adam
	0,999987	7,776E-03	sgd		0,999893	6,522E-04	nadam		1	1,066E-05	adam
M=123	1	1,479E-07	adamax	M=123	1	5,816E-09	adamax	M=123	1	7,152E-09	adam
	1	7,128E-07	adam		1	6,008E-09	adamax		1	8,416E-09	nadam
M=321	1	9,780E-07	adamax	M=321	1	1,478E-05	nadam	M=321	1	2,905E-05	rmsprop
	0,999987	4,517E-05	rmsprop		0,999987	4,549E-05	rmsprop		0,999960	1,946E-04	rmsprop
M=333	1	3,729E-09	adam	M=333	1	2,953E-09	adamax	M=333	1	2,374E-08	adam
	1	2,761E-08	adam		1	5,628E-09	adamax		1	5,544E-08	adam

Рис. 2. Значение характеристик обученных нейросетей из top-2 по тестовой выборке.

На рис. 3 представлено распределение оптимизаторов в top-2 сетях по стратегиям инициализации. На первом месте оказались оптимизаторы *adam* – 14 и *adamax* – 14 (при этом для стратегии **В** первое место занимает *adam*, а для стратегий **А** и **Б** – *adamax*) на втором *nadam* – 9, на третьем *rmsprop* – 6, и *sgd* – 5, а *adagrad* и *adadelat* не оказались среди top-2.

	sgd	rmsprop	adagrad	adadelat	adam	adamax	nadam
А	2	2	0	0	4	6	2
Б	1	2	0	0	1	8	4
В	2	2	0	0	9	0	3

Рис. 3. Распределение оптимизаторов в top-2 нейросетях в зависимости от стратегии инициализации сверточных слоев.

3.3. Устойчивость нейросетей к загрязнению данных

В результате проведенного эксперимента по загрязнению данных по модели (1) были получены графики деградации сетей (зависимость F_1 от уровня α'), представленные на рис.4. На графиках видно, что архитектуры с одной сверткой в последнем сверточном слое (синие и зеленые графики) показывают относительную стабильность с хорошими показателями устойчивости к загрязнению данных, при этом они практически не зависят от стратегии инициализации (за исключением архитектуры $M = 21$ в случае стратегии **В**, показавшей худшие результаты). Тогда как среди архитектур с максимальным количеством сверток в последнем сверточном слое

(оранжевые и красные графики) лучшую устойчивость к загрязнению данным показали top1 сети с инициализацией по стратегии **В**, однако в случае стратегии **А** сети с архитектурой $M = 22$ и $M = 333$ из top-2 показали максимальную устойчивость.

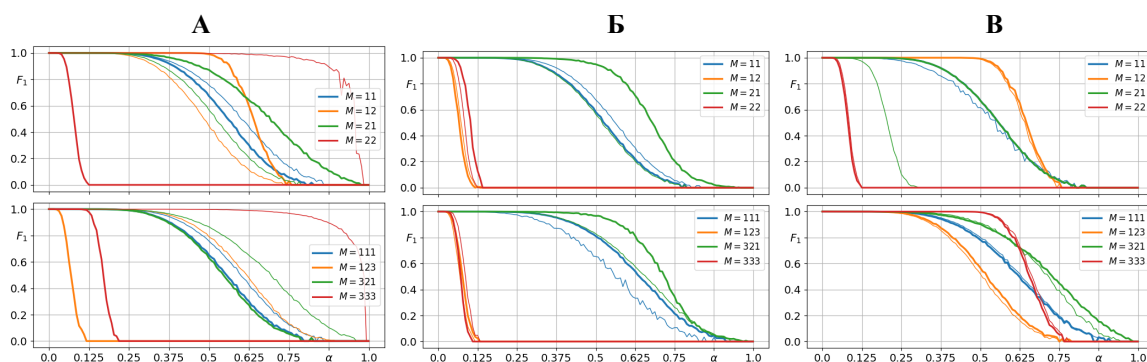


Рис. 4. Оценка устойчивости к загрязнению данных top-2 нейросетей (верхние графики деградации сетей соответствуют двухслойным архитектурам, нижние – трехслойным; по горизонтали представлены графики деградации сетей в соответствии со стратегией инициализации сверточных слоев).

4. Заключение

В работе продемонстрирована проблема «останова» обучения многослойных сверточных нейросетей с малым числом и размером сверток с функцией активации Relu в задаче распознавания шумовых сигналов. Было показано, что инициализация сверточных слоев играет в этом существенную роль. Сравнительный анализ различных стратегий инициализации показал, что предложенные стратегии положительно влияют на обучаемость нейросетей по сравнению со стратегией по умолчанию. Однако было замечено, что количество «хороших» сетей с качеством $F_1 = 1$, а также значения функции потерь L для сетей из top-2, в большей степени зависит не от стратегии инициализации, а от архитектуры, при этом большую роль играет размер последнего сверточного слоя: «чем больше ядер, тем лучше». Также был составлен рейтинг оптимизаторов, с помощью которых были обучены top-2 сети.

Эксперимент по функционированию обученных нейросетей на загрязнённых данных показал, что в условиях априорной неопределенности устойчивость обученных top-2 нейросетей с одной сверткой в последнем сверточном слое практически не зависит от стратегии инициализации сверточных слоев, а сети с более «широкой» архитектурой с инициализацией по стратегии **В** показывают лучшую устойчивость к загрязнению данных.

Список литературы

1. Czubenko M., Kowalczyk Z. A simple neural network for collision detection of collaborative robots // Sensors. 2021. Vol. 21, No. 12. P. 4235.
2. Leng J. et al. M-A3C: a mean-asynchronous advantage actor-critic reinforcement learning method for real-time gait planning of biped robot // IEEE Access. 2022. Vol. 10. P. 76523-76536.
3. Rybczak M., Popowniak N., Lazarowska A. A Survey of Machine Learning Approaches for Mobile Robot Control // Robotics. 2024. Vol. 13, No. 1. P. 12.
4. Cao J., Cao M., Wang J., et al. Urban noise recognition with convolutional neural network // Multimed Tools Appl. 2019. Vol. 78. P. 29021-29041.
5. de Rezende S.W.F., et al. Convolutional neural network and impedance-based SHM applied to damage detection // Engineering Research Express. 2020. Vol. 2, No. 3. P. 035031.

6. de Rezende S.W.F., et al. ISHM for fault condition detection in rotating machines with deep learning models // Journal of the Brazilian Society of Mechanical Sciences and Engineering. 2023. Vol. 45, No. 4. P. 212.
7. Wei W., et al. Research Progress of Noise in High-Speed Cutting Machining // Sensors. 2022. Vol. 22, No. 10. P. 3851.
8. Portsev R.J., Makarenko A.V. Convolutional neural networks for noise signal recognition // Proceedings of MLSP'2018. Aalborg, Denmark: IEEE, 2018. P. 1-6, <http://ieeexplore.ieee.org/document/8516920>.
9. Порцев Р.Ю., Макаренко А.В. Сравнительный анализ функциональных характеристик элементарных двухслойных свёрточных нейросетей в задаче распознавания шумовых сигналов // Труды конференции «Математическая теория управления и ее приложения» (МТУиП-2022, Санкт Петербург). С.Пб.: АО «Концерн «ЦНИИ «Электроприбор», 2022. С. 132-135.
10. Goodfellow I., Bengio Y., Courville A., Deep Learning. Cambridge: MIT press, 2016.
11. Glorot X., Bengio Y. Understanding the difficulty of training deep feedforward neural networks // Proceedings of the thirteenth international conference on artificial intelligence and statistics // JMLR Workshop and Conference Proceedings, 2010. P. 249-256.
12. Keras: The Python Deep Learning library, <https://keras.io/>.
13. TensorFlow is an end-to-end open source platform for machine learning. <https://www.tensorflow.org/>.