© 2021 г. А.В. ГРАБОВОЙ
(Московский физико-технический институт, Москва)
В.В. СТРИЖОВ, д-р физ-мат. наук
(Вычислительный центр имени А. А. Дородницына ФИЦ ИУ РАН)

Байесовская дистилляция моделей глубокого обучения[1]

В данной работе исследуется проблема понижения сложности аппроксимирующих моделей. Рассматриваются методы основаны на дистилляции моделей глубокого обучения. Данные методы основаны на понятии учителя и ученика, где вводится предположение, что модель ученика имеет меньше параметров, чем модель учителя. Предлагается байесовский подход к построению модели ученика, где априорное распределения параметров ученика зависит от апостериорного распределения параметров модели учителя. Авторами предложен метод построение априорного распределения в пространстве параметров модели ученика при помощи суперпозиции локальных преобразований пространства параметров модели учителя. Теоретические результаты об предложенном методе построения априорного распределения параметров ученика анализируются в вычислительном эксперименте на синтетических и реальных данных. В качестве реальных данных рассматривается выборка FashionMNIST.

Ключевые слова: выбор модели; байесовский вывод; дистилляция модели; локальные преобразования; преобразования вероятностных пространств.

1. Введение

В последние несколько лет мощности вычислительных систем выросли (ссылка на статью по выч. тех.). Данный рост мощности вычислительных систем привел к повышению сложности моделей машинного обучения, путем увеличения обучаемых параметров. Примерами таких моделей являются модели: AlexNet [5], VGGNet [6], ResNet [12], BERT [8, 7], mT5 [10], GPT3[9] и другие. В табл. 1 представлено описание

Таблица 1. Число параметров в моделях машинного обучения.

Название	AlexNet	VGGNet	ResNet	BERT	mT5	GPT3
Год	2012	2014	2015	2018	2020	2020
Тип данных	изображение	изображение	изображение	текст	текст	текст
Число параметров, млрд	0,06	0,13	0,06	0,34	13	175

популярных глубоких моделей машинного обучения. Видно, что число параметров

[1]Работа выполнена при поддержке ... (грант №...).

моделей машинного обучения постоянно растет, что влечет снижение интерпретируемости моделей. Данная проблема широко рассматривается в специальном классе задач по Adversarial Attack [4]. Большое число параметров влечет большие требования к вычислительным ресурсам, из-за чего данные модели не могут быть использованы в мобильных устройствах. Для решения данной проблемы распространены методы дистилляции модели [14] с большим числом параметров в модель с малым числом параметров. Дистиллируемая модель с большим числом параметров называется *учитель*, а модель получаемая путем дистилляции называется *ученик*.

Определение 1. Дистилляция модели — уменьшение сложности модели путем выбора модели в множестве более простых моделей на основе параметров и ответов более сложной фиксируемой модели.

Главные идеи дистилляции предложены в работе Дж. Е. Хинтона [14]. В работе предлагается использовать ответы учителя в качестве целевой переменной для обучения модели ученика. Проведено ряд экспериментов, в которых проводилась дистилляции моделей для задачи классификации машинного обучения. Базовый эксперимент на выборке MNIST [15] показал применимость метода для дистилляции избыточно сложной нейросетевой модели в нейросетевую модель меньшей сложности. Эксперимент по распознаванию речи, в котором ансамбль моделей был дистиллирован в одну модель. Также в работе [14] был проведен эксперимент по обучению экспертных моделей на основе одной большой модели при помощи предложенного метода дистилляции.

В работе [3] предложен метод Neuron Selectivity Transfer основаный на минимизации специальной функции потерь основаной на Maximum Mean Discrepancy между выходами всех слоев модели учителя и ученика. В рамках вычислительного эксперимента была показана эффективность данного метода для задачи классификации изображений на примере выборок CIFAR [1] и ImageNet [2].

В данной работе предлагается метод на основе байесовского вывода. В качестве априорного распределения параметров модели ученика предлагается использовать апостериорное распределение параметров модели учителя. Основной проблемой данного подхода является различие в пространствах параметров модели учителя и модели ученика. Авторы предлагают подход основаный на локальных преобразованиях пространств для сопоставления пространств параметров модели ученика и учителя. В результате данного выравнивания параметры модели учителя и модели ученика принадлежат одному пространству и как следствие в качестве априорного распределения параметров модели ученика выбирается апостериорное распределение параметров модели учителя.

В рамках вычислительного эксперимента анализируются методы ...

2. Постановка задачи дистилляции

Задана выборка:

$$\mathbf{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{m}, \qquad \mathbf{x}_i \in \mathbb{R}^n, \quad y_i \mathbb{Y},$$

где $\mathbf{x}_i, y_i$ — признаковое описание и целевая переменная i-го объекта, число объектов вв обучающей выборке обозначается m. Размер признакового описания объектов

обозначим n. Множество $\mathbb{Y} = \{1, \cdots, K\}$ для задачи классификации, где K число классов, множество $\mathbb{Y} = \mathbb{R}$ для задачи регрессии.

Задана модель учителя в виде суперпозиций линейный преобразований и нелинейных преобразований:

$$f = \sigma \circ l\big(\mathbf{U}_T\big) \circ \sigma \circ l\big(\mathbf{U}_{T-1}\big) \circ \cdots \circ \sigma \circ l\big(\mathbf{U}_1\big),$$

где T — число слоев модели учителя, σ — функция активации, а $l\big(\mathbf{U}_t, \mathbf{x}\big) = \mathbf{U}_y\mathbf{x}$ обозначает линейное преобразование. Матрицы $\mathbf{U}_t$ описывают параметры модели учителя f. Каждая матрица $\mathbf{U}_t$ имеет размер $n_t \times n_{t-1}$, где $n_0 = n$, а $n_T = 1$ для задачи регрессии и $n_T = K$ для задачи классификации на K классов. Число параметров p_f нейросетевой модели f вычисляется следующим образом:

$$p_f = \sum_{t=1}^{T} n_t n_{t-1}.$$

Зададим некоторый порядок на множестве параметров модели учителя f. Вектор параметров модели учителя f обозначим $\mathbf{u}$. В данной работе для полносвязнной нейронной сети рассматривается естественный порядок индуцированный номером слоя и номером нейрона и номером элемента вектора параметра нейрона.

Например рассмотрим следующую модель учителя для решения задачи регрессии:

$$f_3 = \sigma\big(\mathbf{U}_3\big(\sigma\big(\mathbf{U}_2\sigma\big(\mathbf{U}_1\mathbf{x}\big)\big)\big)\big),$$

где для рассмотренной модели вектор параметров $\mathbf{u}$ имеет следующий вид:

$$\mathbf{u}_{f_3} = \big[u_1^{1,1}, \cdots, u_1^{1,n}, \cdots, u_1^{n_1,1}, \cdots, u_1^{n_1,n}, u_2^{1,1}, \cdots, u_2^{1,n_1}, \cdots, u_2^{n_2,1}, \cdots, u_2^{n_2,n_1}, u_3^{1,1}, \cdots, u_3^{1,n_2}\big].$$

Пусть для вектора модели учителя f задано апостериорное распределение параметров $p_f\big(\mathbf{u}|\mathbf{D}\big)$.

На основе выборки $\mathbf{D}$ и модели учителя f требуется выбрать модель ученика из параметрического семейства функций:

$$\mathcal{G} = \left\{g | g = \sigma \circ l\big(\mathbf{W}_S\big) \circ \cdots \circ \sigma \circ l\big(\mathbf{W}_1\big), \mathbf{W}_s \in \mathbb{R}^{n_s \times n_{s-1}}\right\},$$

где S число слоев модели ученика, аналогичным образом определяются n_0, n_S и p_g. Аналогичным образом введем вектор параметров модели ученика $\mathbf{w}$. Заметим, что каждая модель g из семейства $\mathcal{G}$ задается своим вектором параметров $\mathbf{w}_g$, а следовательно задача выборка модели $g \in \mathcal{G}$ эквивалента задаче выбора вектора параметров $\mathbf{w} \in \mathbb{R}^{p_g}$.

Выбор оптимальных параметров $\hat{\mathbf{w}} \in \mathbb{R}^{p_g}$ проводится при помощи вариационного вывода на основе совместного правдоподобия модели и данных:

$$\mathcal{L}_{\mathcal{A}}\big(\mathbf{D}, \mathcal{A}\big) = \log p\big(\mathbf{D}|\mathcal{A}\big) = \log \int_{\mathbf{w} \in \mathbb{R}^{p_g}} p\big(\mathbf{D}|\mathbf{w}\big) p\big(\mathbf{w}|\mathcal{A}\big) d\mathbf{w}, \tag{1}$$

где $p_g\big(\mathbf{w}|\mathcal{A}\big)$ — априорное распределение вектора параметров модели ученика. Так как вычисление интеграла (1) является вычислительно сложной задачей, рассмотрим

вариационных подход [24, 25] для решения данной задачи. Пусть задано распределение вариационное распределение параметров модели ученика $q_g\big(\mathbf{w}\big)$, которое аппроксимирует неизвестное апостериорное распределение $p_g\big(\mathbf{w}|\mathbf{D}\big)$. Выбор параметров $\mathbf{w}$ сводится к решению оптимизационной задачи:

$$\hat{\mathbf{w}} = \arg\min_{q_g,\mathbf{w}} \mathrm{D}_{KL}\big(q_g\big(\mathbf{w}\big)||p_g\big(\mathbf{w}|\mathcal{A}\big)\big) - \log p\big(\mathbf{y}|\mathbf{X},\mathbf{w}\big), \tag{2}$$

заметим, что выражение (2) не учитывает модель учителя f при обучении. Для использовании информации об учители предлагается рассмотреть гиперпараметры $\mathcal{A}$ как функцию от апостериорного распределения $p_g\big(\mathbf{u}|\mathbf{D}\big)$.

3. Построение априорного распределения ученика

Пусть апостериорное распределение параметров модели учителя является нормальным распределением:

$$p\big(\mathbf{u}|\mathbf{D}\big) = \mathcal{N}\big(\mathbf{u}_0, \mathbf{\Sigma}_{u_0}\big),$$

где $\mathbf{u}_0$ и $\mathbf{\Sigma}_{u_0}$ параметры апостериорного распределения.

3.1. Учитель и ученик принадлежат одному семейству

Рассмотрим следующие условия:

1) число слоев модели учителя равняется числу слоев модели ученика $S = T$;

2) размеры соответствующих слоев совпадают, другими словами, для всех t, s таких, что $t = s$ выполняется $n_s = n_t$, где n_t обозначает размер t-го слоя учителя, а n_s размер s-го слоя ученика.

В случае выполнения условий 1)–2) априорное распределение модели ученика равняется апостериорному распределения параметров учителя, то есть $p_g\big(\mathbf{u}|\mathbf{D}\big) = p_f\big(\mathbf{u}|\mathbf{D}\big)$.

3.2. Удаление нейрона в слое учителя

Рассмотрим следующие условия:

1) число слоев модели учителя равняется числу слоев модели ученика $S = T$;

2) размеры соответствующих слоев не совпадают, другими словами, для всех t, s таких, что $t = s$ выполняется $n_s \leqslant n_t$, где n_t обозначает размер t-го слоя учителя, а n_s размер s-го слоя ученика.

Проведем адаптацию модели учителя в модель ученика при помощи последовательных преобразований пространства параметров $\mathbf{u}$. Рассмотрим элементарное преобразования:

$$\phi\big(t\big) : \mathbb{R}^{p_f} \to \mathbb{R}^{p_f - 2n_t}$$

вектора $\mathbf{u}$ которое описывает удаление одного нейрона из t-го слоя. Другими словами преобразования $\phi\big(t\big)$ зануляет одну из строк матрицы $\mathbf{U}_t$. Заметим, что данное зануление неизбежно производит зануление соответсвенного столбца матрицы $\mathbf{U}_{t+1}$.

Обозначим новый вектор параметров $\mathbf{u}' = \phi\big(t, \mathbf{u}\big)$, а подномежство элементов, которые были удалены как $\mathbf{u}''$. Аналогичным образом введем обозначения параметров нормального распределения $p\big(\mathbf{u}|\mathbf{D}\big)$, а именно $\mathbf{u}_0^{'}, \mathbf{u}_0^{''}, \mathbf{\Sigma}_{u_0}^{'}, \mathbf{\Sigma}_{u_0}^{''}, \mathbf{\Sigma}_{u_0}^{',''}, \mathbf{\Sigma}_{u_0}^{'','}$.

Распределение параметров после зануления имеет следующий вид:

$$p_{f'}\big(\mathbf{u}'|\mathbf{D}\big) = \mathcal{N}\big(\mathbf{u}'_0 + \mathbf{\Sigma}^{',''}_{u_0}\mathbf{\Sigma}^{''}_{u_0}{}^{-1}\left(\mathbf{0} - \mathbf{u}''_0\right), \mathbf{\Sigma}'_{u_0} - \mathbf{\Sigma}^{',''}_{u_0}\mathbf{\Sigma}^{''}_{u_0}{}^{-1}\mathbf{\Sigma}^{',''}_{u_0}\big), \tag{3}$$

где полученное распределение является оценкой апостериорного распределения модели без одного нейрона. В случае, если отличие в нескольких нейроннах, то выполняется последовательное применение отображения $\phi\big(t, \mathbf{u}\big)$. Заметим, что данный подход имеет ряд недостатков, которые связаны с тем, что данная последовательная процедура является жадной.

3.3. Удаление слоя учителя

Пусть в модели учителя требуется убрать t-й слой. Рассмотрим следующие условия:

1) соответсвующие размеры слоев совпадают $n_t = n_{t-1}$;

2) функция активации удовлетворяет следующему свойству $\sigma \circ \sigma = \sigma$.

Проведем адаптацию модели учителя в модель ученика при помощи последовательных преобразований пространства параметров $\mathbf{u}$. Рассмотрим элементарное преобразования:

$$\psi\big(t\big) : \mathbb{R}^{p_f} \to \mathbb{R}^{p_f - n_t n_{t-1}}$$

вектора $\mathbf{u}$ которое описывает удаление одного t-го слоя. Другими словами преобразования $\psi\big(t\big)$ превращает матрицу $\mathbf{U}_t$ в удиничную матрицу $\mathbf{I}$. Введя аналогичные обозначения как для формулы (3) получим распределение параметров после удаления слоя:

$$p_{f'}\big(\mathbf{u}'|\mathbf{D}\big) = \mathcal{N}\big(\mathbf{u}'_0 + \mathbf{\Sigma}^{',''}_{u_0}\mathbf{\Sigma}^{''}_{u_0}{}^{-1}\left(\mathbf{i} - \mathbf{u}''_0\right), \mathbf{\Sigma}'_{u_0} - \mathbf{\Sigma}^{',''}_{u_0}\mathbf{\Sigma}^{''}_{u_0}{}^{-1}\mathbf{\Sigma}^{',''}_{u_0}\big),$$

где $\mathbf{i} = [\underbrace{1, 0, \cdots, 0}_{n_t}, \underbrace{0, 1, \cdots, 0}_{n_t}, \underbrace{0, 0, 1, \cdots, 0}_{n_t}, \underbrace{0, \cdots, 1}_{n_t}]$ полученное распределение $p\big(\mathbf{u}'|\mathbf{D}\big)$ является оценкой апостериорного распределения модели без одного слоя. В случае, если отличие в нескольких слоях, то выполняется последовательное применение отображения $\psi\big(t, \mathbf{u}\big)$.

3.4. Выполнение последовательных преобразований

Локальные преобразования ϕ, ψ позволяют провести выравнивание пространств параметров учителя f и ученика g. После выравнивания пространств параметров в качестве априорного распределения параметров ученика $p_g\big(\mathbf{w}|\mathbf{D}\big)$ рассматривается полученное апостериорное распределение параметров учителя $p_{f'}\big(\mathbf{u}'|\mathbf{D}\big)$. Получаем, следующее приближение априорного распределения:

$$p_g\big(\mathbf{w}|\mathbf{D}\big) = p_{f'}\big(\mathbf{w}|\mathbf{D}\big),$$

где данное априорное распределение использования для поиска оптимальных параметров модели ученика $\hat{\mathbf{w}}$ используя выражение (2).

4. Вычислительный эксперимент

5. Заключение

СПИСОК ЛИТЕРАТУРЫ

1. *Alex Krizhevsky and Vinod Nair and Geoffrey Hinton* CIFAR-10 (Canadian Institute for Advanced Research) // `http://www.cs.toronto.edu/~kriz/cifar.html`

2. *Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.* Imagenet: A large-scale hierarchical image database // IEEE conference on computer vision and pattern recognition, 2009. P. 248–255.

3. *Huang, Zehao and Wang, Naiyan* Like What You Like: Knowledge Distill via Neuron Selectivity Transfer // arXiv e-prints, 2017.

4. *Kui Ren and Tianhang Zheng and Zhan Qin and Xue Liu* Adversarial Attacks and Defenses in Deep Learning // Engineering, 2020. P. 346–360.

5. *Alex Krizhevsky, Ilya Sutskever, Geoffrey Hinton* ImageNet Classification with Depp Convolutional Neural Networks // NIPS, 2012.

6. *Karen Simonyan and Andrew Zisserman* Very Deep Convolutional Networks for Large-Scale Image Recognition // NIPS, 2014.

7. *Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A., Kaiser L., Polosukhin I.* Attention Is All You Need // In Advances in Neural Information Processing Systems. 2017. V. 5. P. 6000–6010.

8. *Devlin J., Chang M., Lee K., Toutanova K.* BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding // arXiv preprinted, 2018.

9. *Tom B. Brown et al* GPT3: Language Models are Few-Shot Learners // arXiv preprinted, 2020.

10. *Linting Xue and Noah Constant and Adam Roberts and Mihir Kale and Rami Al-Rfou and Aditya Siddhant and Aditya Barua and Colin Raffel.* mT5: A massively multilingual pre-trained text-to-text transformer // arXiv preprinted, 2021.

11. *Yang, Ziqing and Cui, Yiming and Chen, Zhipeng and Che, Wanxiang and Liu, Ting and Wang, Shijin and Hu, Guoping* TextBrewer: An Open-Source Knowledge Distillation Toolkit for Natural Language Processing // Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations. 2020. P. 9–16.

12. *He K., Zhang X., Ren S., Sun J.* Deep Residual Learning for Image Recognition // Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016. P. 770–778.

13. *Бахтеев О. Ю., Стрижов В. В.* Выбор моделей глубокого обучения субоптимальной сложности // АиТ. 2018. № 8. С. 129–147.

14. *Hinton G., Vinyals O., Dean J.* Distilling the Knowledge in a Neural Network // NIPS Deep Learning and Representation Learning Workshop. 2015.

15. *LeCun Y., Cortes C., Burges C.* The MNIST dataset of handwritten digits, 1998. http://yann.lecun.com/exdb/mnist/index.html.

16. *Vapnik V., Izmailov R.* Learning Using Privileged Information: Similarity Control and Knowledge Transfer // Journal of Machine Learning Research. 2015. No 16. P. 2023–2049.

17. *Lopez-Paz D., Bottou L., Scholkopf B., Vapnik V.* Unifying Distillation and Privileged Information // In International Conference on Learning Representations. Puerto Rico, 2016.

18. *Madala H., Ivakhnenko A.* Inductive Learning Algorithms for Complex Systems Modeling. Boca Raton: CRC Press Inc., 1994.

19. *Xiao H., Rasul K.,Vollgraf R.* Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms // arXiv preprint arXiv:1708.07747. 2017.

20. *Wilson T., Kozareva Z., Nakov P., Rosenthal S., Stoyanov V., Ritter A.* SemEval-2013 Task 2: Sentiment Analysis in Twitter // Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013). Atlanta, 2013. P. 312–320.

21. *LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W., Jackel L.* Backpropagation Applied to Handwritten Zip Code Recognition // Neural Computation. 1989. V. 1. No 4. P. 541–551.

22. *Hochreiter S., Schmidhuber J.* Long short-term memory // Neural Computation. 1997. V. 9. No 8. P. 1735–1780.

23. *Kingma D, Ba J.* Adam: A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980. 2014.

24. *Graves A.* Practical Variational Inference for Neural Networks // Advances in Neural Information Processing Systems, 2011. Vol. 24. P. 2348–2356.

25. *Grabovoy A.V., Bakhteev O.Y., Strijov V.V.* Estimation of relevance for neural network parameters // Informatics and Applications, 2019. Vol.13 No 2. P. 62–70.