Revisiting deep learning research

(and deep learning for tabular data)

Иван Рубачёв

Кто я?

- Учился на ПМИ в 2016-2020
  - Большой любитель НИСа
  - курсач, практика, диплом
- Поработал год в яндексе
- Год назад перешел в research

https://twitter.com/irubachev

https://github.com/puhsu

https://t.me/puhsuuu

План

1. Про проблемы

   Несколько статей про честность и воспроизводимость

   - semi-supervised learning reality check (2018)
   - few-shot learning (2019)
   - metric-learning reality checks (2020)
   - neural rankers (2021)

2. Про нашу статью которая не добавляет проблем

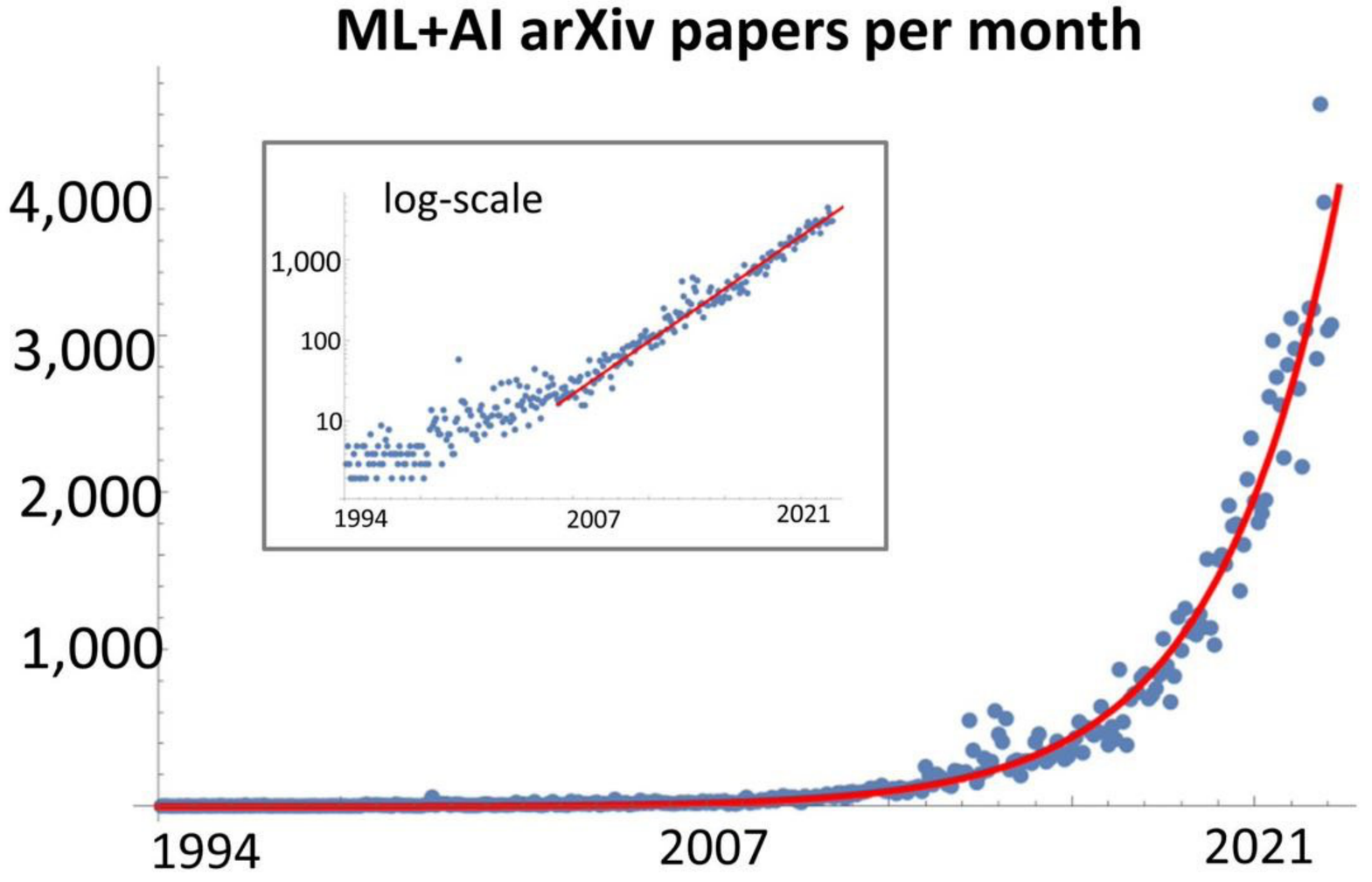
   Что нового придумали в deep learning’е для работы с табличными данными

   Правда ли что всё* это не работает?

*почти всё

Revisiting dl research

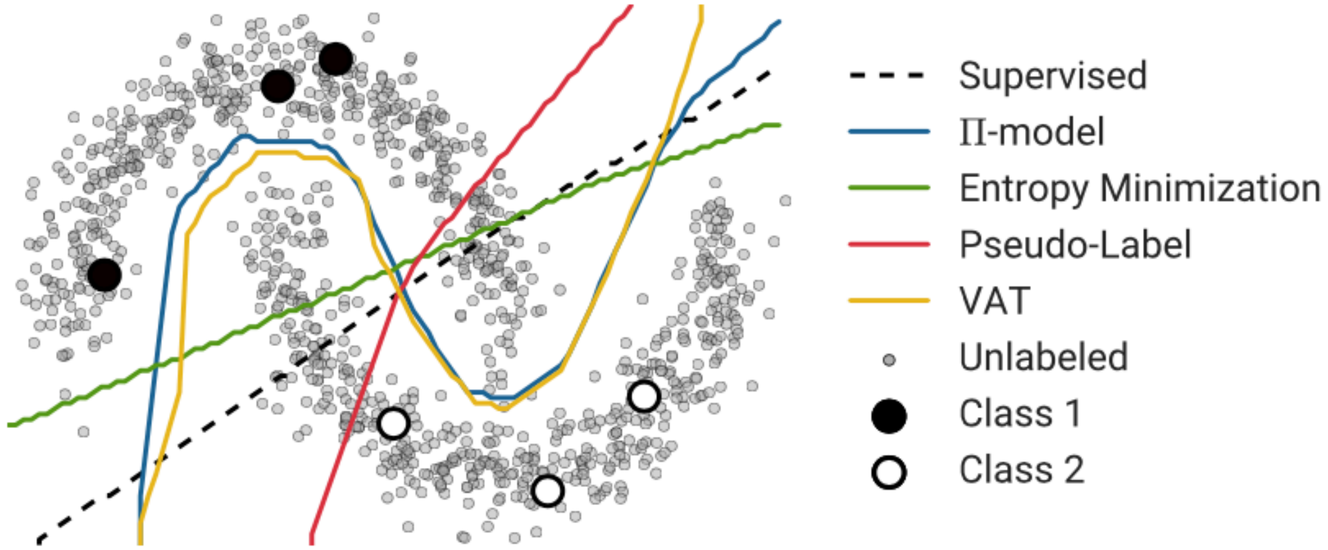
```
curl arxiv.org/ml_papers.txt | grep “revisiting *”
```



https://twitter.com/mariokrenn6240/status/1430556920390332416

Realistic Evaluation of Deep Semi-Supervised Learning Algorithms

https://arxiv.org/abs/1804.09170



Realistic Evaluation of Deep Semi-Supervised Learning Algorithms

https://arxiv.org/abs/1804.09170

Method	CIFAR-10 4000 Labels	SVHN 1000 Labels
Π-Model [32]	34.85% → 12.36%	19.30% → 4.80%
Π-Model [46]	13.60% → 11.29%	–
Π-Model (ours)	20.26% → 16.37%	12.83% → 7.19%
Mean Teacher [50]	20.66% → 12.31%	12.32% → 3.95%
Mean Teacher (ours)	20.26% → 15.87%	12.83% → 5.65%

"We find the gap between the fully-supervised baseline and those obtained with SSL is smaller in our study than what is generally reported in the literature"

Realistic Evaluation of Deep Semi-Supervised Learning Algorithms

https://arxiv.org/abs/1804.09170

Method	CIFAR-10 4000 Labels	SVHN 1000 Labels
Π-Model [32]	34.85% → 12.36%	19.30% → 4.80%
Π-Model [46]	13.60% → 11.29%	–
Π-Model (ours)	20.26% → 16.37%	12.83% → 7.19%
Mean Teacher [50]	20.66% → 12.31%	12.32% → 3.95%
Mean Teacher (ours)	20.26% → 15.87%	12.83% → 5.65%

"Can we design a model which can match the performance of SSL techniques without using any unlabeled data?..."

*...this model obtained an average test error of **13.4%** over 5 independent runs. This result emphasises the importance of the underlying model in the evaluation of SSL algorithms, and reinforces our point that **different algorithms must be evaluated using the same model to avoid conflating comparison**."*

Realistic Evaluation of Deep Semi-Supervised Learning Algorithms

https://arxiv.org/abs/1804.09170

Realistic evaluation of deep semi-supervised learning algorithms

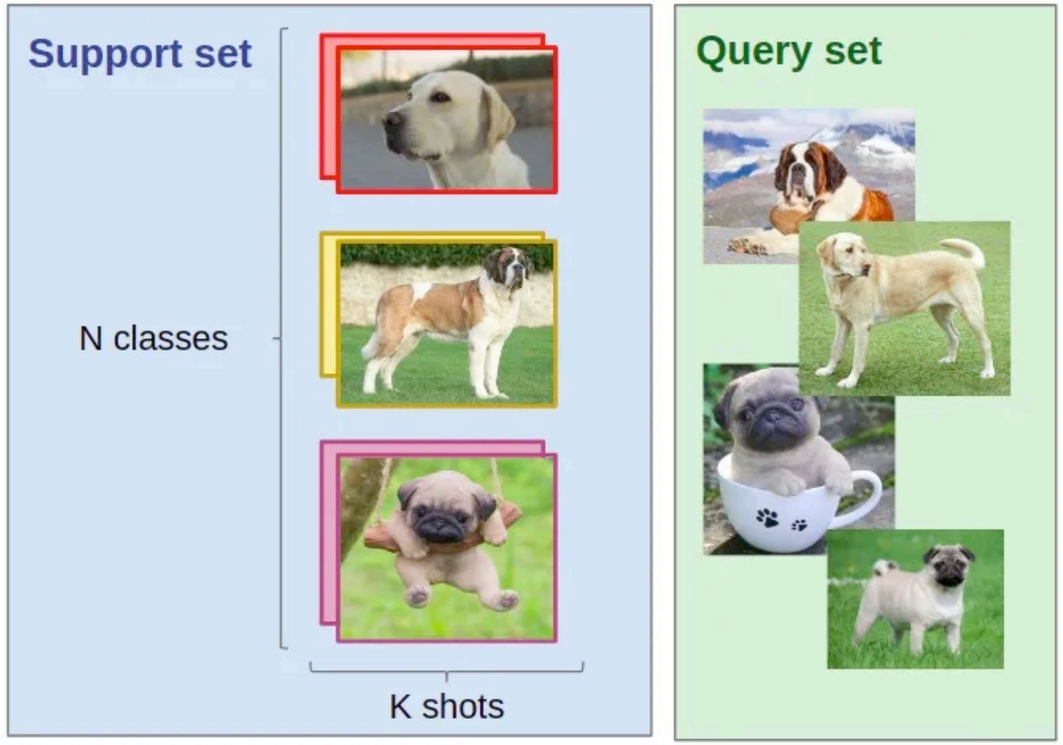
A Oliver, A Odena, C Raffel, ED Cubuk… - arXiv preprint arXiv …, 2018 - arxiv.org

Semi-supervised learning (SSL) provides a powerful framework for leveraging unlabeled data when labels are limited or expensive to obtain. SSL algorithms based on deep neural networks have recently proven successful on standard benchmark tasks. However, we argue that these benchmarks fail to address many issues that these algorithms would face in real-world applications. After creating a unified reimplementation of various widely-used SSL techniques, we test them in a suite of experiments designed to address these issues …

☆ 99 Цитируется: 513 Похожие статьи Все версии статьи (9) »

SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning

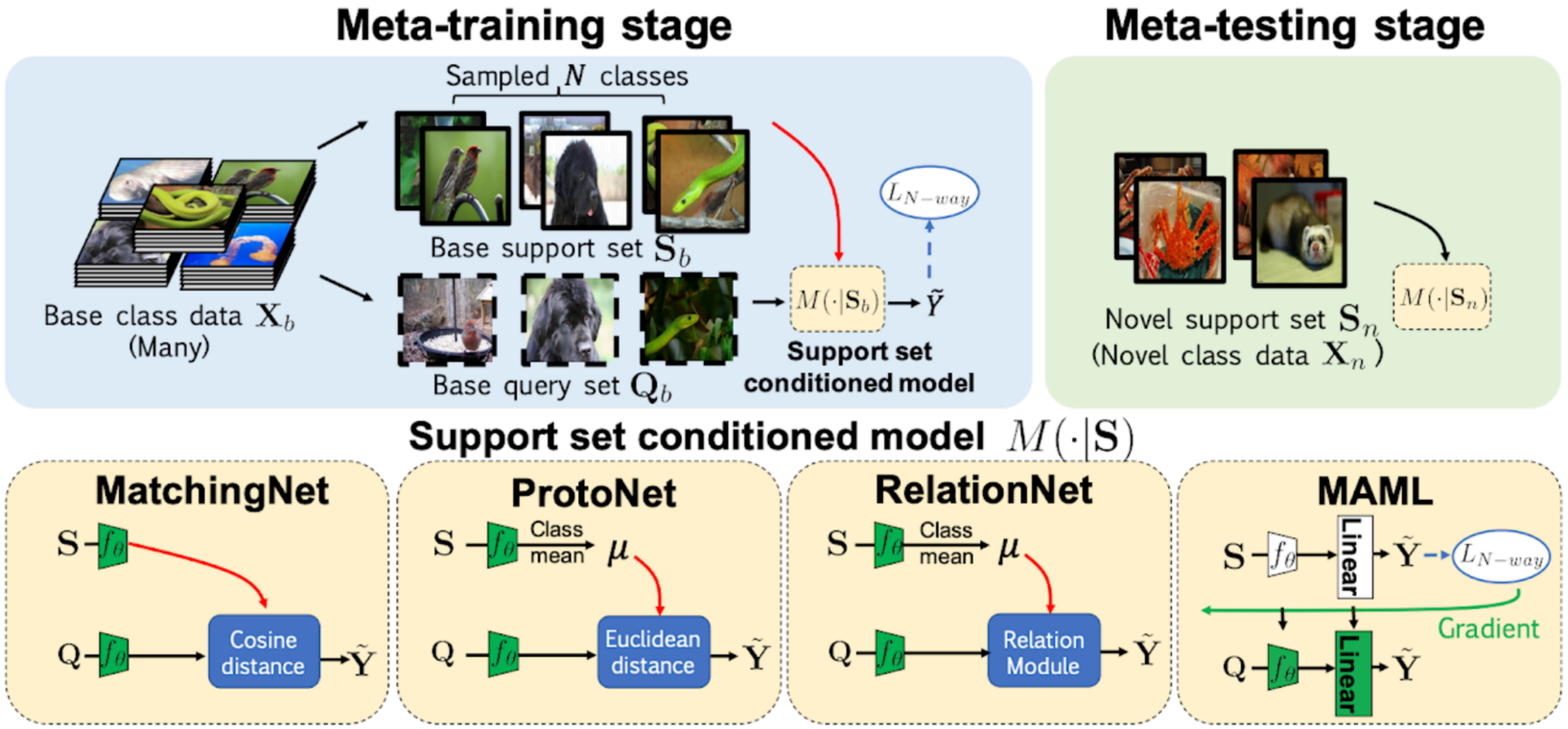
https://arxiv.org/abs/1911.04623



https://neptune.ai/blog/understanding-few-shot-learning-in-computer-vision

SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning

https://arxiv.org/abs/1911.04623



https://sites.google.com/view/a-closer-look-at-few-shot/

SimpleShot: Revisiting Nearest-Neighbor Classification for Few-Shot Learning

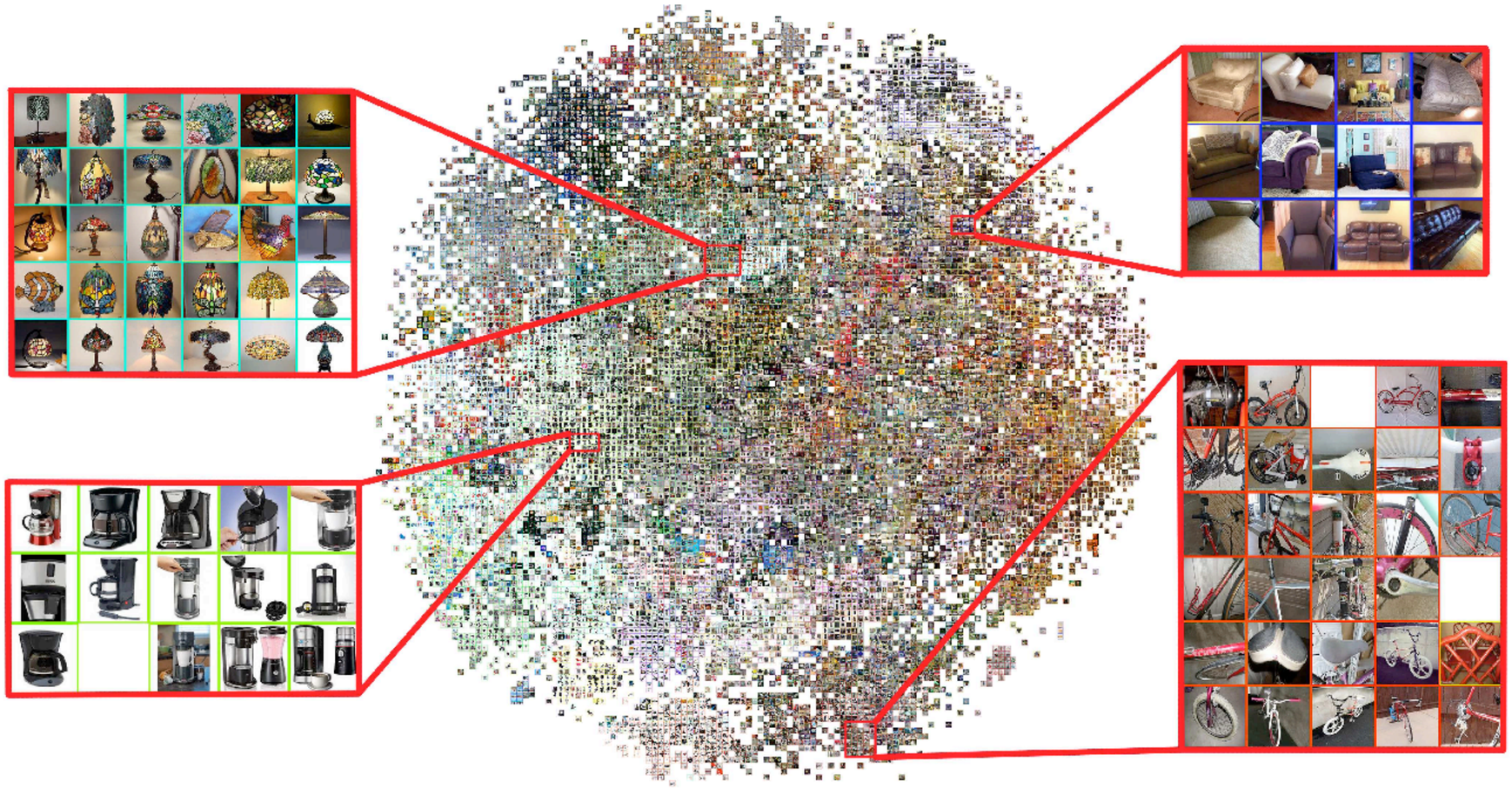
https://arxiv.org/abs/1911.04623

Table 1: Average accuracy (in %; measured over 600/10,000 rounds*) of one-shot and five-shot classifiers for five-way classification on *mini*ImageNet; higher is better. The best result of each network architecture of each column is in **bold** font. Results of our approaches are in blue. Best viewed in color.

Approach	Network	One shot	Five shots
Meta LSTM [26]	Conv-4	43.44 ± 0.77	60.60 ± 0.71
MatchingNet [34]	Conv-4	43.56 ± 0.84	55.31 ± 0.73
MAML [4]	Conv-4	48.70 ± 1.84	63.11 ± 0.92
LLAMA [10]	Conv-4	49.40 ± 1.83	–
ProtoNet [30]	Conv-4	49.42 ± 0.78	68.20 ± 0.66
Reptile [23]	Conv-4	49.97 ± 0.32	65.99 ± 0.58
PLATIPUS [5]	Conv-4	50.13 ± 1.86	–
mAP-SSVM [32]	Conv-4	50.32 ± 0.80	63.94 ± 0.72
GNN [6]	Conv-4	50.33 ± 0.36	66.41 ± 0.63
RelationNet [31]	Conv-4	50.44 ± 0.82	65.32 ± 0.70
Meta SGD [18]	Conv-4	50.47 ± 1.87	64.03 ± 0.94
MTNet [17]	Conv-4	51.70 ± 1.84	–
Qiao *et al.* [25]	Conv-4	54.53 ± 0.40	67.87 ± 0.20
FEAT [36]	Conv-4	**55.15 ± 0.20**	**71.61 ± 0.16**
SimpleShot (UN)	Conv-4	33.17 ± 0.17	63.25 ± 0.17
SimpleShot (L2N)	Conv-4	48.08 ± 0.18	66.49 ± 0.17
SimpleShot (CL2N)	Conv-4	49.69 ± 0.19	66.92 ± 0.17
MAML [4]†	ResNet-18	49.61 ± 0.92	65.72 ± 0.77
Chen *et al.* [2]	ResNet-18	51.87 ± 0.77	75.68 ± 0.63
RelationNet [31]†	ResNet-18	52.48 ± 0.86	69.83 ± 0.68
MatchingNet [34]†	ResNet-18	52.91 ± 0.88	68.88 ± 0.69
ProtoNet [30]†	ResNet-18	54.16 ± 0.82	73.68 ± 0.65
Gidaris *et al.* [8]	ResNet-15	55.45 ± 0.89	70.13 ± 0.68
SNAIL [21]	ResNet-15	55.71 ± 0.99	68.88 ± 0.92
Bauer *et al.* [1]	ResNet-34	56.30 ± 0.40	73.90 ± 0.30
adaCNN [22]	ResNet-15	56.88 ± 0.62	71.94 ± 0.57
TADAM [24]	ResNet-15	58.50 ± 0.30	76.70 ± 0.30
CAML [15]	ResNet-12	59.23 ± 0.99	72.35 ± 0.71
SimpleShot (UN)	ResNet-10	54.45 ± 0.21	76.98 ± 0.15
SimpleShot (L2N)	ResNet-10	57.85 ± 0.20	78.73 ± 0.15
SimpleShot (CL2N)	ResNet-10	60.85 ± 0.20	78.40 ± 0.15
SimpleShot (UN)	ResNet-18	56.06 ± 0.20	78.63 ± 0.15
SimpleShot (L2N)	ResNet-18	60.16 ± 0.20	79.94 ± 0.14
SimpleShot (CL2N)	ResNet-18	**62.85 ± 0.20**	**80.02 ± 0.14**

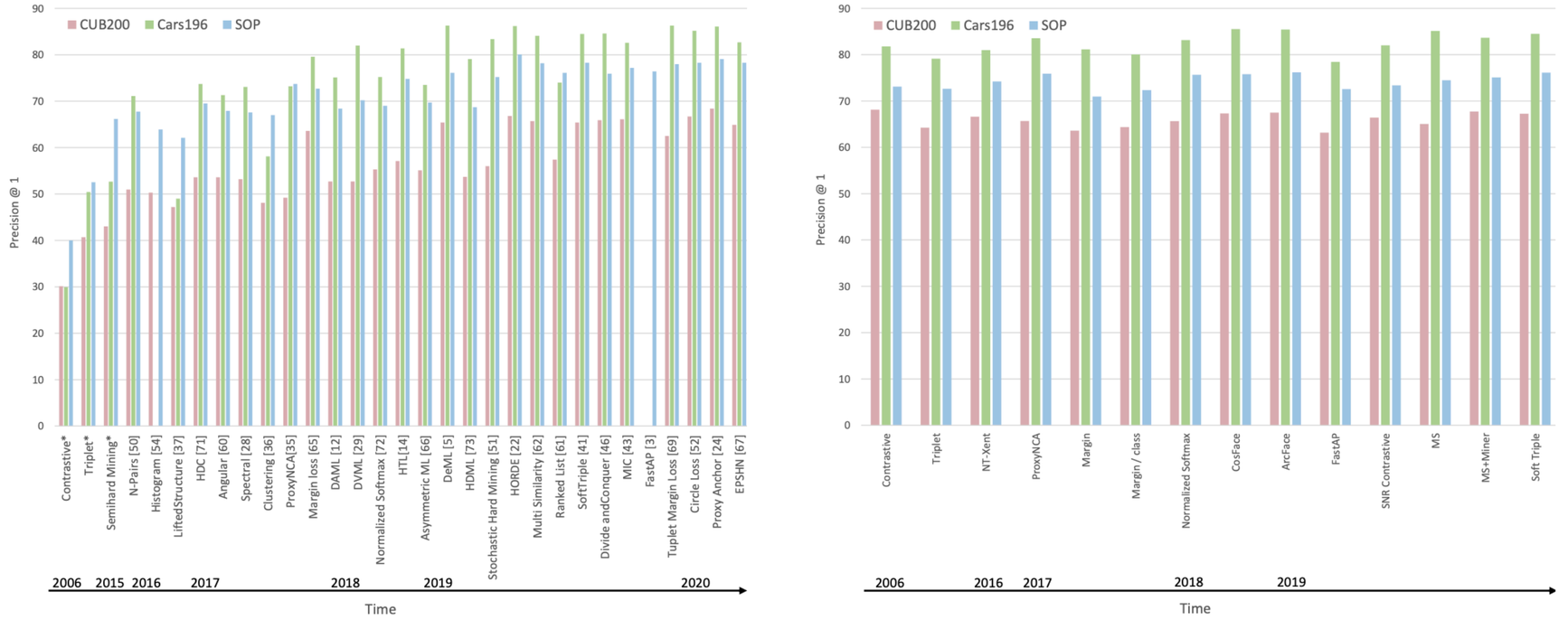
A Metric Learning Reality Check

https://arxiv.org/abs/2003.08505



A Metric Learning Reality Check

https://arxiv.org/abs/2003.08505



(a) The trend according to papers

(b) The trend according to reality

Fig. 2. Papers versus Reality: the trend of Precision@1 of various methods over the years. In a), the baseline methods have * next to them, which indicates that their numbers are the average reported accuracy from all papers that included those baselines.

Are Neural Rankers still Outperformed by Gradient Boosted Decision Trees?

https://iclr.cc/virtual/2021/spotlight/3536

- Нет стандартного протокола сравнения
- Слабый GBDT бейзлайн
- Нет аккуратности

Are Neural Rankers still Outperformed by Gradient Boosted Decision Trees?

https://iclr.cc/virtual/2021/spotlight/3536

Table 2: All numbers are significantly worse than the corresponding number from λMART_{GBM} at the $p < 0.05$ level using a two-tailed t-test. Best performing numbers are bold.

Models	Rerank	Web30K NDCG@k			Yahoo NDCG@k			Istella NDCG@k		
		@1	@5	@10	@1	@5	@10	@1	@5	@10
$\lambda\text{MART}_{RankLib}$	✗	45.35	44.59	46.46	68.52	70.27	74.58	65.71	61.18	65.91
λMART_{GBM}	✗	**50.73**	**49.66**	**51.48**	**71.88**	**74.21**	**78.02**	**74.92**	**71.24**	**76.07**
RankSVM	✗	30.10	33.50	36.50	63.70	67.40	72.60	52.69	50.41	55.29
GSF	✗	41.29	41.51	43.74	64.29	68.38	73.16	62.24	59.68	65.08
ApproxNDCG	✗	46.64	45.38	47.31	69.63	72.32	76.77	65.81	62.32	67.09
DLCM	✓	46.30	45.00	46.90	67.70	69.90	74.30	65.58	61.94	66.80
SetRank	✗	42.90	42.20	44.28	67.11	69.60	73.98	67.33	62.78	67.37
SetRank^{re}	✓	45.91	45.15	46.96	68.22	70.29	74.53	67.60	63.45	68.34

"In this paper, we showed the inconsistency of performance comparison between neural rankers and GBDT models, and ***verified the inferior performance of neural models****..."*

Are Neural Rankers still Outperformed by Gradient Boosted Decision Trees?

https://iclr.cc/virtual/2021/spotlight/3536

Table 3: Result on the Web30K, Yahoo, and Istella datasets. $^{\uparrow}$ means significantly better result, performanced against λMART_{GBM} at the $p < 0.05$ level using a two-tailed t-test. Last row is relative difference of DASALC-ens over λMART_{GBM}.

Models	Web30K NDCG@k			Yahoo NDCG@k			Istella NDCG@k		
	@1	@5	@10	@1	@5	@10	@1	@5	@10
λMART_{GBM}	50.73	49.66	51.48	**71.88**	**74.21**	**78.02**	**74.92**	71.24	76.07
SetRank^{re}	45.91	45.15	46.96	68.22	70.29	74.53	67.60	63.45	68.34
DASALC	50.95	50.92$^{\uparrow}$	52.88$^{\uparrow}$	70.98	73.76	77.66	72.77	70.06	75.30
DASALC-ens	**51.89**$^{\uparrow}$	**51.72**$^{\uparrow}$	**53.73**$^{\uparrow}$	71.24	74.07	77.97	74.40	**71.32**	**76.44**$^{\uparrow}$
(Relative diff)	(+2.29%)	(+4.15%)	(+4.37%)	(-0.89%)	(-0.18%)	(-0.06%)	(-0.69%)	(+0.11%)	(+0.49%)

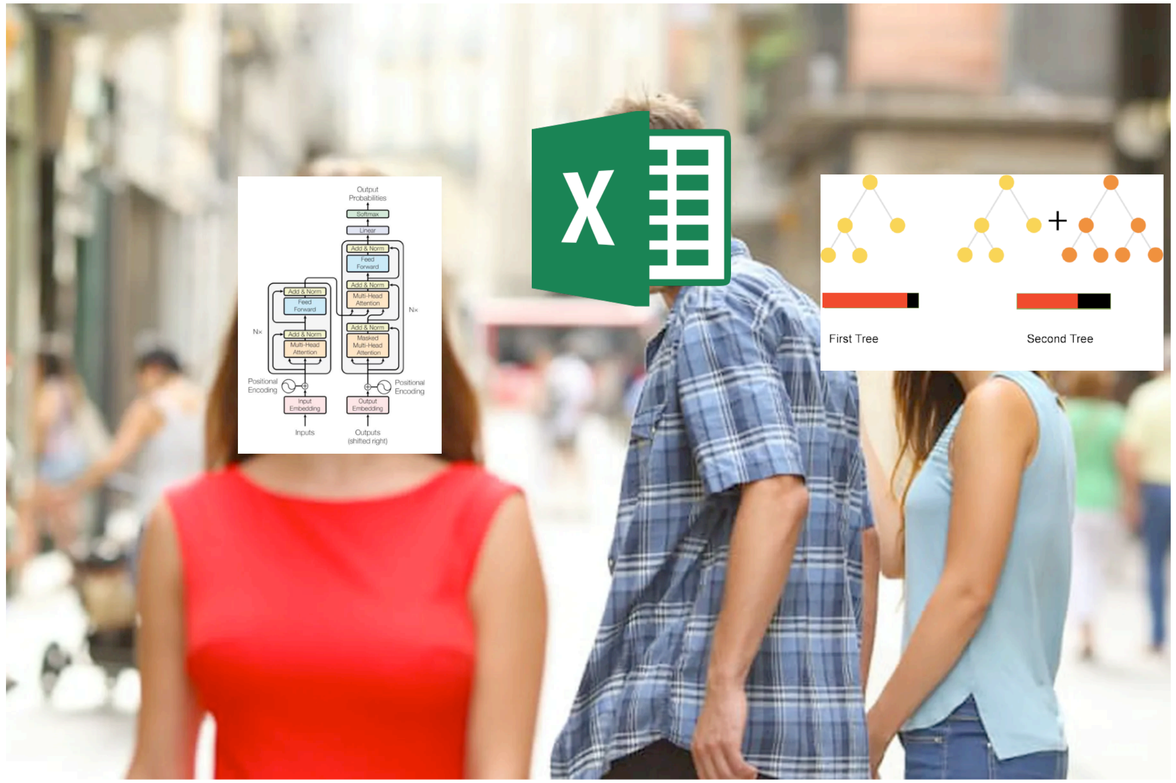
"...We identified the weaknesses when building neural rankers in multiple components and proposed methods to address them. Our proposed framework performs competitively well with the ***strong tree-based baselines****"*

“Если не знаешь про что написать статью, выбери любую область и наведи в ней порядок”

(c)

Revisiting dl for tabular data

How it started



How it's going

Revisiting Deep Learning Models for Tabular Data

Yury Gorishniy[*†‡] **Ivan Rubachev**[†♣] **Valentin Khrulkov**[†] **Artem Babenko**[†♣]

† Yandex, Russia
‡ Moscow Institute of Physics and Technology, Russia
♣ National Research University Higher School of Economics, Russia

arXiv.org > cs > arXiv:1706.09516

Computer Science > Machine Learning

[Submitted on 28 Jun 2017 (v1), last revised 20 Jan 2019 (this version, v5)]

CatBoost: unbiased boosting with categorical features

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, Andrey Gulin

arXiv.org > cs > arXiv:1909.06312

Computer Science > Machine Learning

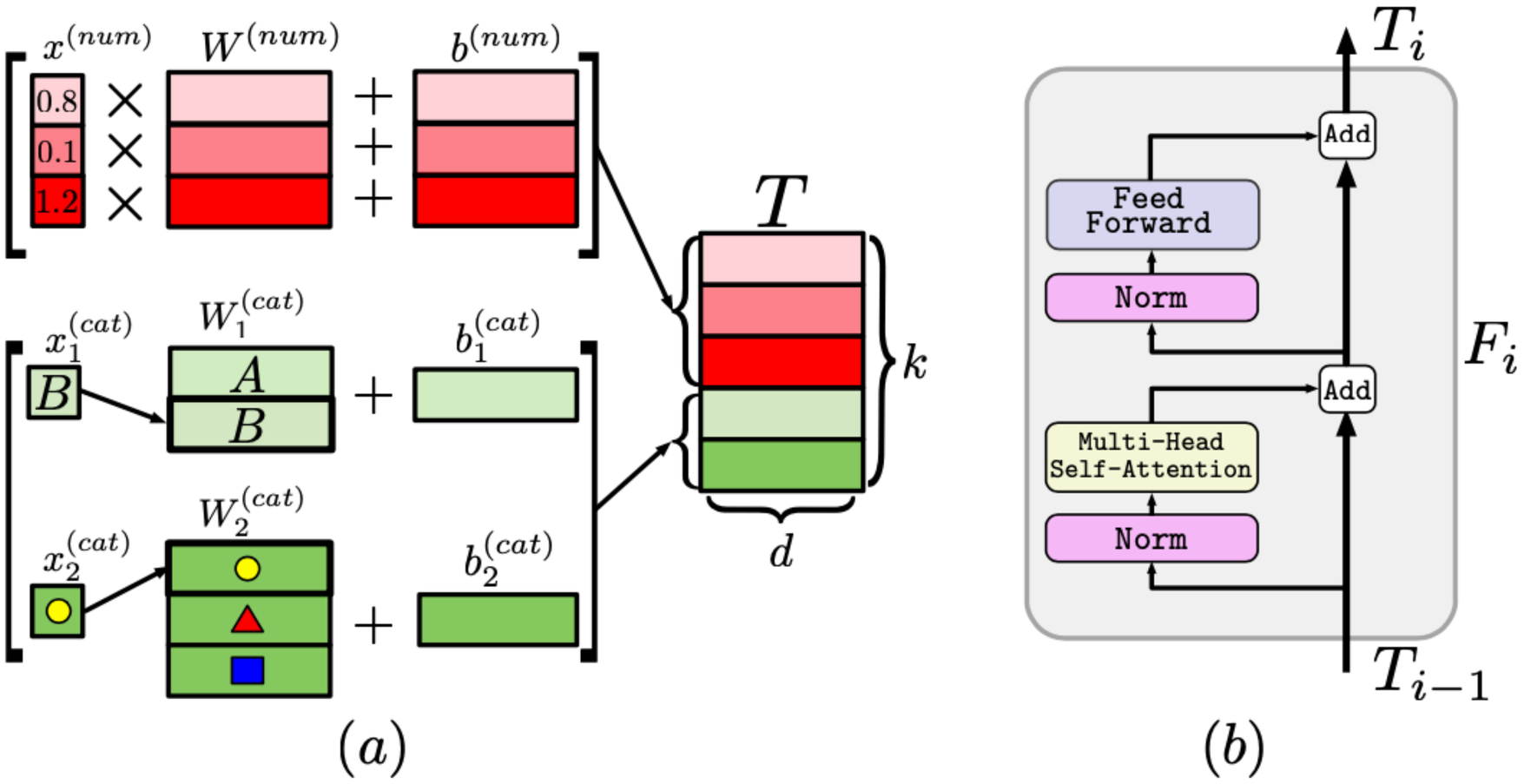
[Submitted on 13 Sep 2019 (v1), last revised 19 Sep 2019 (this version, v2)]

Neural Oblivious Decision Ensembles for Deep Learning on Tabular Data

Sergei Popov, Stanislav Morozov, Artem Babenko

• • •

Transformer



Transformer

	CA ↓	AD ↑	HE ↑	JA ↑	HI ↑	AL ↑	EP ↑	YE ↓	CO ↑	YA ↓	MI ↓
FT-Transformer											
FT-Transformer$_d$	0.455	0.801	0.3948	0.735	0.730	0.966	0.8969	8.719	**0.9695**	**0.748**	**0.7429**
FT-Transformer	**0.450**	0.810	**0.3983**	**0.737**	0.731	**0.967**	0.8984	8.722	0.9692	**0.748**	0.7434
GBDT											
CatBoost$_d$	0.428	0.798	0.3863	0.724	0.726	0.948	0.8894	8.885	0.9096	0.7490	0.7440
CatBoost	**0.423**	0.794	0.3885	0.727	0.726	–	0.8899	8.837	0.9685	0.7401	**0.7413**
XGBoost$_d$	0.463	0.775	0.3502	0.721	0.705	0.925	0.8803	9.446	0.9640	0.7732	0.7719
XGBoost	0.431	0.796	0.3767	0.725	0.725	–	0.8880	8.819	**0.9696**	**0.7320**	0.7421

Baselines

TabNet: Attentive Interpretable Tabular Learning

Sercan O. Arik, Tomas Pfister

We propose a novel high-performance and interpretable canonical deep tabular data learning architecture, TabNet. TabNet uses sequential attention to choose which features to reason from at each decision step, enabling interpretability and more efficient learning as the learning capacity is used for the most salient features. We demonstrate that TabNet outperforms other neural network and decision tree variants on a wide range of non-performance-saturated tabular datasets and yields interpretable feature attributions plus insights into the global model behavior. Finally, for the first time to our knowle
supervised learning for tabular data, significantly improving performance with
learning when unlabeled data is abundant.

Gradient Boosting Neural Networks: GrowNet

Sarkhan Badirli, Xuanqing Liu, Zhengming Xing, Avradeep Bhowmik, Khoa Doan, Sathiya S. Keerthi

A novel gradient boosting framework is proposed where shallow neural networks are employed as ``weak learners''. General loss functions are considered under this unified framework with specific examples presented for classification, regression, and learning to rank. A fully corrective step is incorporated to remedy the pitfall of greedy function approximation of classic gradient boosting decision tree. The proposed model rendered outperforming results against state-of-the-art boosting methods in all three tasks on multiple datasets. An ablation study is performed to shed light on the effect of each model components and

SAINT: Improved Neural Networks for Tabular Data via Row Attention and Contrastive Pre-Training

Gowthami Somepalli, Micah Goldblum, Avi Schwarzschild, C. Bayan Bruss, Tom Goldstein

Tabular data underpins numerous high-impact applications of machine learning from fraud detection to genomics and healthcare. Classical approaches to solving tabular problems, such as gradient boosting and random forests, are widely used by practitioners. However, recent deep learning methods have achieved a degree of performance competitive with popular techniques. We devise a hybrid deep learning approach to solving tabular data problems. Our method, SAINT, performs attention over both rows and columns, and it includes an enhanced embedding method. We also study a new contrastive self-supervised pre-training method for use when labels are scarce. SAINT consistently improves performance over previous deep learning methods, and it even outperforms gradient boosting methods, including XGBoost, CatBoost, and LightGBM, on average over a variety of benchmark tasks.

- Нет общего бенчмарка (ImageNet, GLUE)
- GBDT много раз побежден, но новые модели почему-то есть только в статьях

Наводим порядок

Общий бенчмарк:

- собрали 11 больших табличных датасетов
- Одинаковый бюджет на тюнинг параметров
- Общий код тюнинга, обучения и теста
- Много запусков

В итоге:

- Сильный бейзлайн (ResNet)
- Бейзлайн еще сильнее (Transformer)
- GBDT не побеждены

Наводим порядок

	CA ↓	AD ↑	HE ↑	JA ↑	HI ↑	AL ↑	EP ↑	YE ↓	CO ↑	YA ↓	MI ↓
Baseline Neural Networks											
SNN	0.507	**0.816**	0.3728	0.718	0.721	0.954	0.8970	8.881	0.9465	0.769	0.7521
TabNet	0.513	0.796	0.3782	0.724	0.717	0.954	0.8902	9.032	0.9335	0.819	0.7565
GrowNet	0.500	0.793	–	–	0.724	–	**0.8977**	8.866	–	0.775	0.7549
DCN2	0.486	0.784	0.3853	0.714	0.720	0.955	**0.8975**	8.939	0.9491	0.766	0.7500
AutoInt	0.479	0.801	0.3722	0.716	**0.726**	0.945	0.8948	8.875	0.9312	0.795	0.7517
MLP	0.494	0.796	0.3832	0.719	0.721	0.954	0.8968	8.861	0.9499	0.776	0.7521
NODE	**0.464**	0.791	0.3593	**0.726**	0.724	0.918	0.8958	**8.774**	0.9436	**0.762**	**0.7474**
ResNet	0.487	**0.816**	**0.3960**	**0.727**	**0.727**	**0.963**	0.8971	8.845	**0.9560**	0.766	0.7493
FT-Transformer											
FT-Transformer$_d$	0.470	0.799	0.3812	0.725	0.723	0.953	0.8959	8.869	0.9617	**0.758**	**0.7475**
FT-Transformer	**0.464**	0.807	0.3913	**0.731**	**0.728**	0.960	**0.8982**	8.820	**0.9641**	**0.758**	**0.7469**
GBDT											
CatBoost$_d$	**0.430**	0.797	0.3814	0.721	0.724	0.946	0.8882	8.913	0.9076	0.751	0.7454
CatBoost	**0.431**	0.791	0.3853	0.723	0.725	–	0.8880	8.877	0.9658	0.743	0.7429
XGBoost$_d$	0.463	0.775	0.3502	0.721	0.705	0.925	0.8803	9.446	0.9640	0.773	0.7719
XGBoost	0.433	0.796	0.3755	0.724	0.725	–	0.8857	8.947	**0.9695**	**0.736**	**0.7424**

When FT-Transformer is better than ResNet?

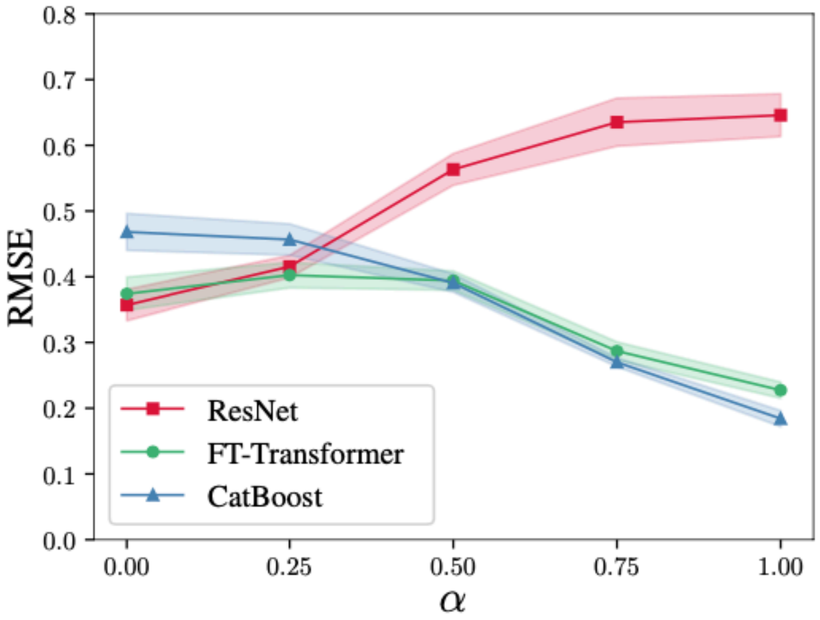


Figure 3: RMSE for the same test set with five different regression targets. The results are averaged over five seeds, shadows represent standard deviations.

$$x \sim \mathcal{N}(0, I_k),$$
$$y = \alpha \cdot f_{GBDT}(x) + (1 - \alpha) \cdot f_{DNN}(x).$$

Итог

Статей предлагающих yet another крутой метод много (и меньше не будет)

Пишите/читайте более важные статьи

Хотел добавить, но не успел/не понял куда

- Importance of deconstruction talk https://slideslive.com/38938218/the-importance-of-deconstruction
- Please commit more academic fraud https://jacobbuckman.com/2021-05-29-please-commit-more-blatant-academic-fraud/
- ICML -> ICCV plagiarism https://www.reddit.com/r/MachineLearning/comments/p59pzp/d_imitation_is_the_sincerest_form_of_flattery/
- Revisiting training in metric learning https://arxiv.org/abs/2002.08473
- Revisiting resnets https://arxiv.org/abs/2103.07579
- Object detection training strategies https://arxiv.org/abs/2107.00057