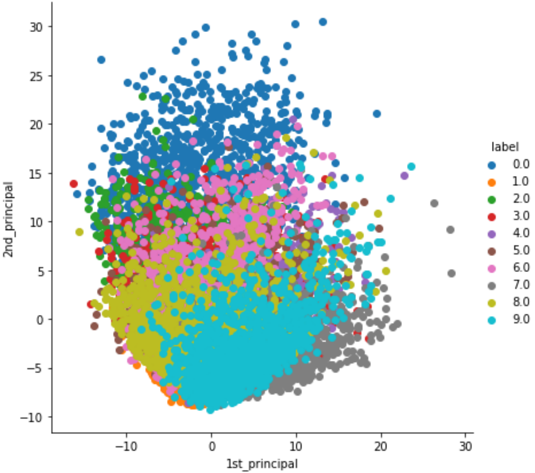
Проекция в латентное пространство

Московский Физико-Технический Институт

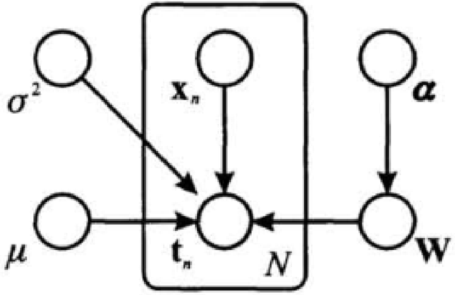
2022

Метод главных компонент

$$\boldsymbol{W} = \arg\max Var(\boldsymbol{X}\boldsymbol{W})$$



Bayesian PCA

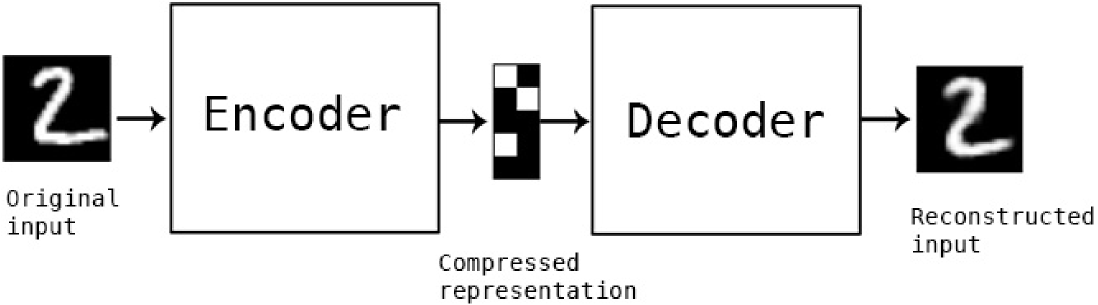


Автокодировщик

Автокодировщик — модель снижения размерности:

$$\boldsymbol{H} = \boldsymbol{\sigma}(\boldsymbol{W}_e \boldsymbol{X}),$$

$$||\boldsymbol{\sigma}(\boldsymbol{W}_d \boldsymbol{H}) - \boldsymbol{X}||_2^2 \to \min .$$



Многообразие

Определение, wiki

Многообразие — пространство, локально сходное с евклидовым.

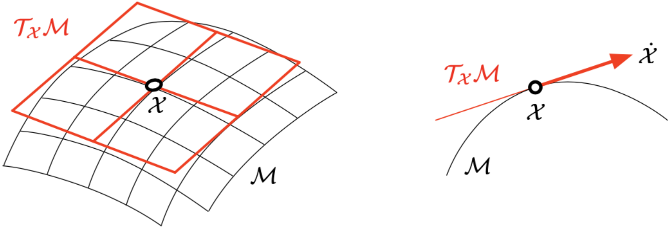
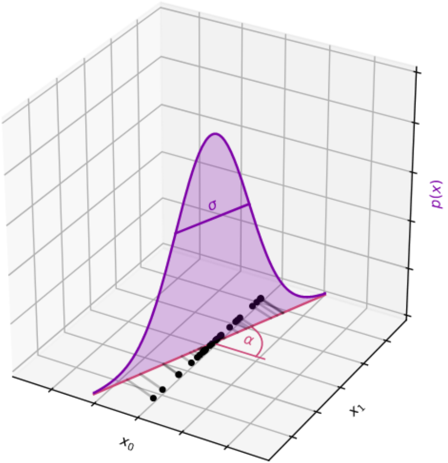
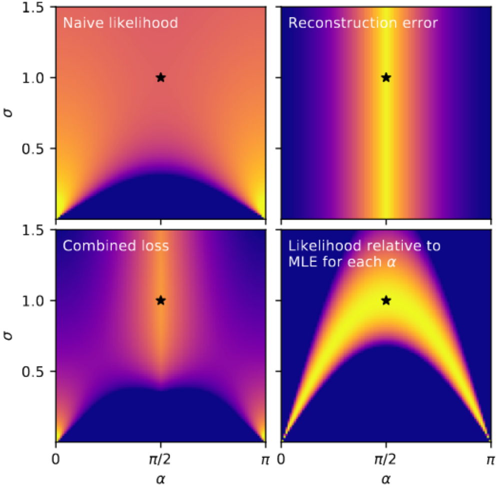


Figure 2. A manifold $\mathcal{M}$ and the vector space $T_\mathcal{X}\mathcal{M}$ (in this case $\cong \mathbb{R}^2$) tangent at the point $\mathcal{X}$, and a convenient side-cut. The velocity element, $\dot{\mathcal{X}} = \partial\mathcal{X}/\partial t$, does not belong to the manifold $\mathcal{M}$ but to the tangent space $T_\mathcal{X}\mathcal{M}$.

Многообразие: зачем нужно?





Автокодировщик: порождающая модель?

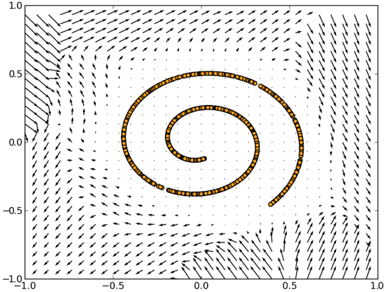
(Alain, Bengio 2012): рассмотрим модель автокодировщика с регуляризацией:

$$||\boldsymbol{f}(\boldsymbol{x}, \sigma) - \boldsymbol{x}||^2,$$

где σ — уровень шума, подаваемого на вход модели кодирования. Тогда

$$\frac{\partial \log p(x)}{\partial x} = \frac{||\boldsymbol{f}(\boldsymbol{x}, \sigma) - \boldsymbol{x}||^2}{\sigma^2} + o(1) \text{ при } \sigma \to 0.$$

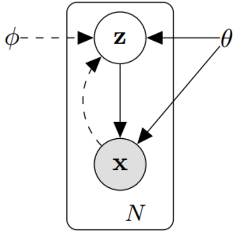
Векторное поле, индуцированное ошибкой реконструкции автокодировщика



Вариационный автокодировщик

Пусть объекты выборки $\boldsymbol{X}$ порождены при условии скрытой переменной $\boldsymbol{h} \sim \mathcal{N}(0, \boldsymbol{I})$:

$$\boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{h}, \boldsymbol{w}).$$



$p(\boldsymbol{h}|\boldsymbol{x}, \boldsymbol{w})$ — неизвестно.
Будем максимизировать вариационную оценку правдоподобия выборки:

$$\log p(\boldsymbol{x}|\boldsymbol{w}) \geq \mathsf{E}_{q_\phi(\boldsymbol{h}|\boldsymbol{x})} \log p(\boldsymbol{x}|\boldsymbol{h}, \boldsymbol{w}) - D_{\mathrm{KL}}(q_\phi(\boldsymbol{h}|\boldsymbol{x}) || p(\boldsymbol{h})) \to \max.$$

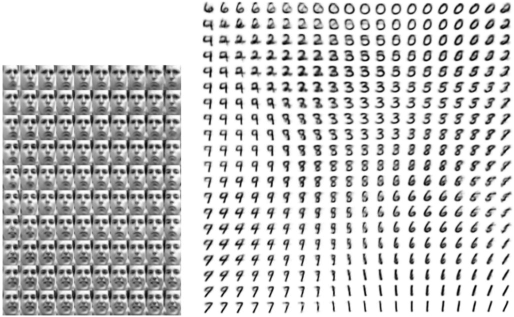
Распределения $q_\phi(\boldsymbol{h}|\boldsymbol{x})$ и $p(\boldsymbol{x}|\boldsymbol{h}, \boldsymbol{w})$ моделируются нейросетью:

$$q_\phi(\boldsymbol{h}|\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}_\phi(\boldsymbol{x}), \boldsymbol{\sigma}^2_\phi(\boldsymbol{x})),$$

$$p(\boldsymbol{x}|\boldsymbol{h}, \boldsymbol{w}) \sim \mathcal{N}(\boldsymbol{\mu}_w(\boldsymbol{h}), \boldsymbol{\sigma}^2_w(\boldsymbol{h})),$$

Вариационный автокодировщик: процесс порождения

Процесс порождения заключается в сэмплировании скрытой переменной из априорного расспределения: $\boldsymbol{z} \sim p(\boldsymbol{z})$ и действии на него декодером.



(a) Learned Frey Face manifold (b) Learned MNIST manifold

Несколько пространств

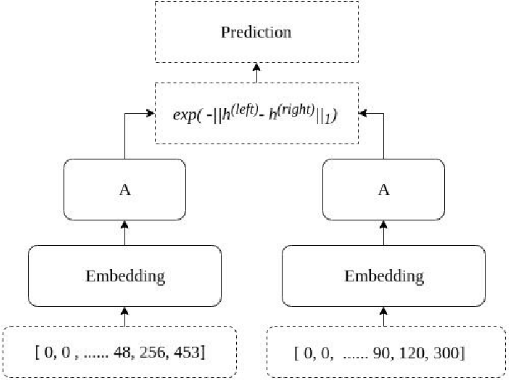
Заданы два пространства $\boldsymbol{X}, \boldsymbol{Y}$.
Как построить общее латнетное пространство между ними?

Несколько пространств

Заданы два пространства $\boldsymbol{X}, \boldsymbol{Y}$.
Как построить общее латнетное пространство между ними?

Наивный метод: $||f(\boldsymbol{X}) - f(\boldsymbol{Y})|| \to \min$ не работает, нужна регуляризация.

Сиамские сети



Обучение метрики

$$D(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{(\boldsymbol{x}_1 - \boldsymbol{x}_2)^\mathsf{T} \boldsymbol{M} (\boldsymbol{x}_1 - \boldsymbol{x}_2)}$$

Триплетные ограничения

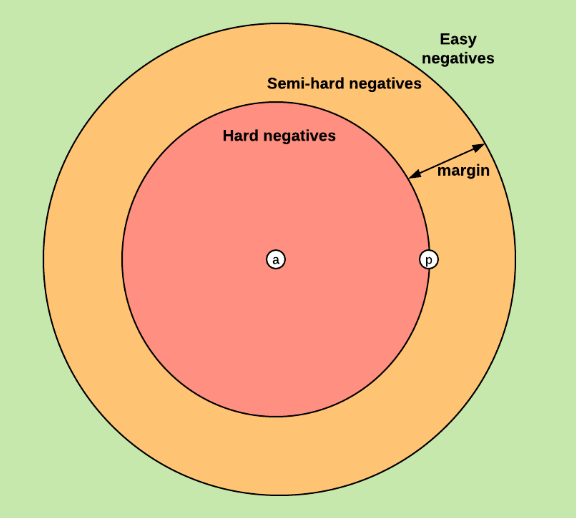
The loss function for each sample in the mini-batch is:

$$L(a, p, n) = \max\{d(a_i, p_i) - d(a_i, n_i) + \mathrm{margin}, 0\}$$

where

$$d(x_i, y_i) = \left\|\mathbf{x}_i - \mathbf{y}_i\right\|_p$$

Триплетные ограничения



Bayesian representation learning with oracle constraints

$$p(t_{i,j,l}) = \int_z p(t_{i,j,l}|z_i, z_j, z_l)p(\boldsymbol{z}_i)p(\boldsymbol{z}_j)p(\boldsymbol{z}_k)dz_i dz_j dz_k,$$

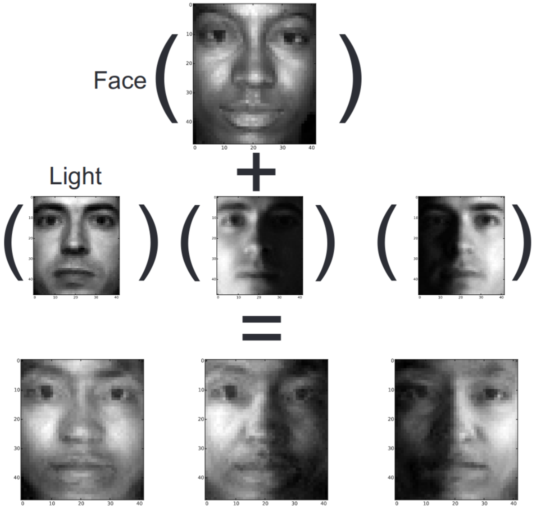
this gives the following likelihood:

$$p(t_{i,j,l}) = Ber(t_{i,j,l}) = \frac{\mathrm{e}^{-D_{i,j}}}{\mathrm{e}^{-D_{i,j}} + \mathrm{e}^{-D_{i,l}}}$$

with

$$D_{a,b} = \sum_{h=1}^{H} D_{a,b}^{h} = -\sum_{h=1}^{H} \left[\mathrm{JS}\left(p(\boldsymbol{z}_a^h) || p(\boldsymbol{z}_b^h) \right) \right].$$

Bayesian representation learning with oracle constraints



Variational learning across domains with triplet information

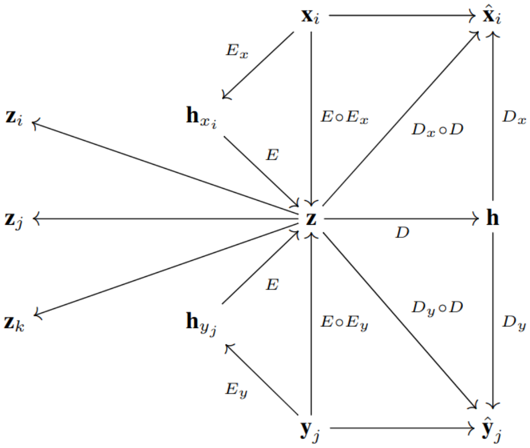


Figure 1: VBTA generative process

Variational learning across domains with triplet information

$$\mathcal{L}_{VBTA} = \mathbb{E}_{q_{\phi_x}(\mathbf{z}_x|\mathbf{x})} \log \frac{p_{\theta_x}(\mathbf{x}, \mathbf{y}, \mathbf{t}, \mathbf{z}_x)}{q_{\phi_x}(\mathbf{z}_x|\mathbf{x})} + \mathbb{E}_{q_{\phi_y}(\mathbf{z}_y|\mathbf{y})} \log \frac{p_{\theta_y}(\mathbf{x}, \mathbf{y}, \mathbf{t}, \mathbf{z}_y)}{q_{\phi_y}(\mathbf{z}_y|\mathbf{y})} =$$

$$= \underbrace{-\Big[KL\big(q_{\phi_{\mathbf{x}}(\mathbf{z}_x|\mathbf{x})}(\mathbf{z}_x|\mathbf{x}) \parallel p_{\theta_{\mathbf{x}}}(\mathbf{z}_x)\big) + KL\big(q_{\phi_{\mathbf{y}}(\mathbf{z}_y|\mathbf{y})}(\mathbf{z}_y|\mathbf{y}) \parallel p_{\theta_{\mathbf{y}}}(\mathbf{z}_y)\big)\Big]}_{\text{Penalty}} +$$

$$+ \underbrace{\Big[\mathbb{E}_{q_{\phi_{\mathbf{x}}}(\mathbf{z}_x|\mathbf{x})}\big[\log p_{\theta_{\mathbf{x}}}(\mathbf{x}|\mathbf{z}_x)\big] + \mathbb{E}_{q_{\phi_{\mathbf{y}}}(\mathbf{z}_y|\mathbf{y})}\big[\log p_{\theta_{\mathbf{y}}}(\mathbf{y}|\mathbf{z}_y)\big]\Big]}_{\text{Reconstruction}} +$$

$$+ \underbrace{\Big[\mathbb{E}_{q_{\phi_{\mathbf{x}}}(\mathbf{z}_x|\mathbf{x})}\big[\log p_{\theta_{\mathbf{x}}}(\mathbf{y}|\mathbf{z}_x)\big] + \mathbb{E}_{q_{\phi_{\mathbf{y}}}(\mathbf{z}_y|\mathbf{y})}\big[\log p_{\theta_{\mathbf{y}}}(\mathbf{x}|\mathbf{z}_y)\big]\Big]}_{\text{Cycle-consistency}} +$$

$$+ \underbrace{\mathbb{E}_{q_{\phi_{\mathbf{x}}}(\mathbf{z}_x|\mathbf{x})}\big[\log p(\mathbf{t}|\mathbf{z}_x)\big] + \mathbb{E}_{q_{\phi_{\mathbf{y}}}(\mathbf{z}_y|\mathbf{x})}\big[\log p(\mathbf{t}|\mathbf{z}_y)\big]}_{\text{Triplet likelihood}}$$

Differentiable Neural Architecture Search in Equivalent Space with Exploration Enhancement

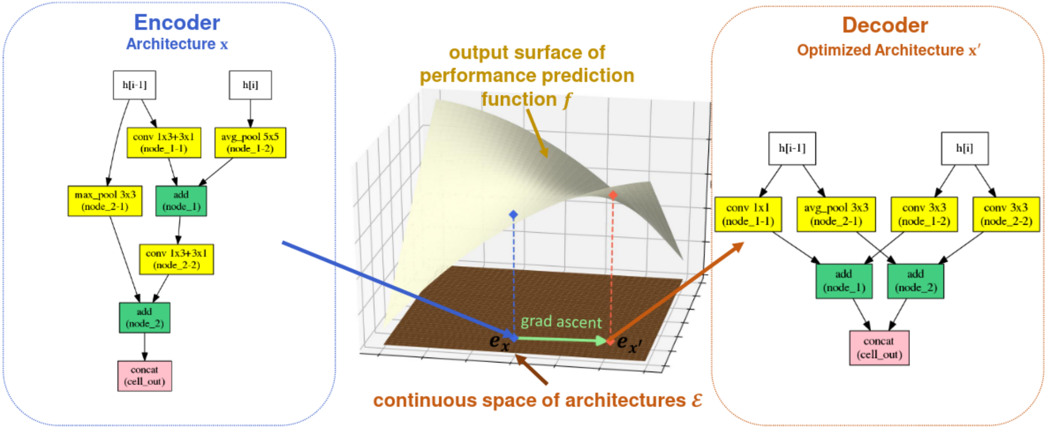
- Для представления структуры используется графовый supervised-автокодировщик
- Оптимизация структуры: DARTS + поощрение новых структур

Table 1: Comparison results with state-of-the-art NAS approaches on NAS-Bench-201.

Method	CIFAR-10		CIFAR-100		ImageNet-16-120	
	Valid(%)	Test(%)	Valid(%)	Test(%)	Valid(%)	Test(%)
ENAS	37.51±3.19	53.89±0.58	13.37±2.35	13.96±2.33	15.06±1.95	14.84±2.10
RandomNAS*	85.63±0.44	88.58±0.21	60.99±2.79	61.45±2.24	31.63±2.15	31.37±2.51
DARTS (1st)	39.77±0.00	54.30±0.00	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
DARTS (2nd)	39.77±0.00	54.30±0.00	15.03±0.00	15.61±0.00	16.43±0.00	16.32±0.00
SETN	84.04±0.28	87.64±0.00	58.86±0.06	59.05±0.24	33.06±0.02	32.52±0.21
NAO*	82.04±0.21	85.74±0.31	56.36±3.14	59.64±2.24	30.14±2.02	31.35±2.21
GDAS*	90.03±0.13	93.37±0.42	70.79±0.83	70.35±0.80	40.90±0.33	41.11±0.13
E^2NAS	**90.94±0.83**	**93.89±0.47**	**71.83±1.84**	**72.05±1.58**	**45.44±1.24**	**45.77±1.00**

Neural Architecture Optimization, 2019

- Для представления структуры используется LSTM



Does Unsupervised Architecture Representation Learning Help Neural Architecture Search?, 2020

- Для представления структуры используется графовый вариационный автокодировщик
- Оптимизация в ддва этапа: unsupervised для представлений моделей, затем — supervised (RL + BO)

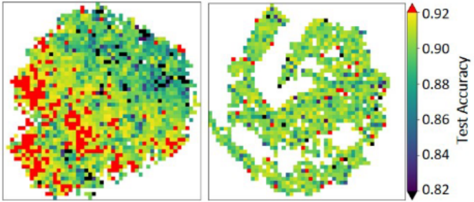


Figure 4: Latent space 2D visualization [65] comparison between *arch2vec* (left) and supervised architecture representation learning (right) on NAS-Bench-101. Color encodes test accuracy. We randomly sample $10,000$ points and average the accuracy in each small area.

Does Unsupervised Architecture Representation Learning Help Neural Architecture Search?, 2020

NAS Methods	CIFAR-10		CIFAR-100		ImageNet-16-120	
	validation	test	validation	test	validation	test
RE [44]	91.08±0.43	93.84±0.43	73.02±0.46	72.86±0.55	45.78±0.56	45.63±0.64
RS [66]	90.94±0.38	93.75±0.37	72.17±0.64	72.05±0.77	45.47±0.65	45.33±0.79
REINFORCE [10]	91.03±0.33	93.82±0.31	72.35±0.63	72.13±0.79	45.58±0.62	45.30±0.86
BOHB [12]	90.82±0.53	93.61±0.52	72.59±0.82	72.37±0.90	45.44±0.70	45.26±0.83
arch2vec-RL	91.32±0.42	94.12±0.42	73.13±0.72	73.15±0.78	46.22±0.30	46.16±0.38
arch2vec-BO	**91.41±0.22**	**94.18±0.24**	**73.35±0.32**	**73.37±0.30**	**46.34±0.18**	**46.27±0.37**

Table 3: The mean and standard deviation of the validation and test accuracy of different algorithms under three datasets in NAS-Bench-201. The results are calculated over 500 independent runs.

	Test Error		Params (M)	Search Cost				
NAS Methods	Avg	Best		Stage 1	Stage 2	Total	Encoding	Search Method
Random Search [15]	3.29±0.15	-	3.2	-	-	4	-	Random
ENAS [68]	-	2.89	4.6	0.5	-	-	Supervised	REINFORCE
ASHA [69]	3.03±0.13	2.85	2.2	-	-	9	-	Random
RS WS [69]	2.85±0.08	2.71	4.3	2.7	6	8.7	-	Random
SNAS [16]	2.85±0.02	-	2.8	1.5	-	-	Supervised	GD
DARTS [15]	2.76±0.09	-	3.3	4	1	5	Supervised	GD
BANANAS [49]	2.64	2.57	3.6	100 (queries)	-	11.8	Supervised	BO
Random Search (ours)	3.1±0.18	2.71	3.2	-	-	4	-	Random
DARTS (ours)	2.71±0.08	2.63	3.3	4	1.2	5.2	Supervised	GD
BANANAS (ours)	2.67±0.07	2.61	3.6	100 (queries)	1.3	11.5	Supervised	BO
arch2vec-RL	2.65±0.05	2.60	3.3	100 (queries)	1.2	9.5	Unsupervised	REINFORCE
arch2vec-BO	**2.56±0.05**	**2.48**	3.6	100 (queries)	1.3	10.5	Unsupervised	BO

Table 4: Comparison with state-of-the-art cell-based NAS methods on DARTS search space using CIFAR-10. The test error is averaged over 5 seeds. Stage 1 shows the GPU days (or number of queries) for model search and Stage 2 shows the GPU days for model evaluation.

Литература

- Bishop C. M. Pattern recognition //Machine learning. – 2006. – Т. 128. – №. 9.
- Bishop C. Bayesian pca //Advances in neural information processing systems. – 1998. – Т. 11.
- Sola J., Deray J., Atchuthan D. A micro Lie theory for state estimation in robotics //arXiv preprint arXiv:1812.01537. – 2018.
- Brehmer J., Cranmer K. Flows for simultaneous manifold learning and density estimation //Advances in Neural Information Processing Systems. – 2020. – Т. 33. – С. 442-453.
- Alain G., Bengio Y. What regularized auto-encoders learn from the data-generating distribution //The Journal of Machine Learning Research. – 2014. – Т. 15. – №. 1. – С. 3563-3593.
- Kingma D. P., Welling M. Auto-encoding variational bayes //arXiv preprint arXiv:1312.6114. – 2013.
- Ranasinghe T., Orăsan C., Mitkov R. Semantic textual similarity with siamese neural networks //Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019). – 2019. – С. 1004-1011.
- https://russianblogs.com/article/5172713037/
- Karaletsos T., Belongie S., Rätsch G. Bayesian representation learning with oracle constraints //arXiv preprint arXiv:1506.05011. – 2015.
- Kuznetsova R., Bakhteev O., Ogaltsov A. Variational learning across domains with triplet information //arXiv preprint arXiv:1806.08672. – 2018.
- Zhang M. et al. Differentiable neural architecture search in equivalent space with exploration enhancement //Advances in Neural Information Processing Systems. – 2020. – Т. 33. – С. 13341-13351.
- Luo R. et al. Neural architecture optimization //Advances in neural information processing systems. – 2018. – Т. 31.
- Yan S. et al. Does unsupervised architecture representation learning help neural architecture search? //Advances in Neural Information Processing Systems. – 2020. – Т. 33. – С. 12486-12498.