История развития искусственного интеллекта: вклад Джеффри Хинтона, нобелевского лауреата и «отца глубокого обучения»

Воронцов Константин Вячеславович

д.ф.-м.н., профессор РАН
зав. кафедрой машинного обучения и цифровой гуманитаристики МФТИ
зав. кафедрой математических методов прогнозирования ВМК МГУ
зав. лабораторией МОСА Института искусственного интеллекта МГУ
г.н.с. ФИЦ ИУ РАН

• Нобелевские лекции • Минск, Беларусь • 10 февраля 2025 •

Содержание

1. **Машинное обучение и нейронные сети**
   - Эмпирическая индукция
   - Нейронные сети
   - Вехи и этапы развития
2. **Архитектуры и методы обучения**
   - Сети ассоциативной памяти
   - От машин Больцмана к генеративным моделям
   - Метод обратного распространения ошибок
3. **Глубокие нейронные сети**
   - Глубина важнее ширины
   - Свёрточные нейронные сети
   - Капсульные нейронные сети

Принцип эмпирической индукции

«Не следует полагаться на сформулированные аксиомы и формальные базовые понятия, какими бы привлекательными и справедливыми они не казались. Законы природы нужно «расшифровывать» из фактов опыта. Следует искать правильный метод анализа и обобщения опытных данных; здесь логика Аристотеля не подходит в силу её абстрактности, оторванности от реальных процессов и явлений.»

Фрэнсис Бэкон
(1561–1626)

Таблица открытия: множество объектов $\{x_i : i = 1, \ldots, \ell\}$

- $f(x_i)$ — измеряемое значение *признака* объекта x_i
- $y_i \in \mathbb{R}$ — измеряемое значение *целевого свойства* x_i, либо $y_i \in \{0, 1\}$ — отсутствие или наличие *целевого свойства*

Фрэнсис Бэкон. Новый органон. 1620.

Задача восстановления зависимостей по эмпирическим данным

Дано: $\{x_i : i = 1, \ldots, \ell\}$ — обучающая выборка объектов
$f_j(x)$ — признаки объекта x, $j = 1, \ldots, n$
$y_i = y(x_i)$ — ответы (значения целевого свойства y), $i = 1, \ldots, \ell$

Найти: параметры w модели $a(x, w)$, приближающей зависимость $y(x)$

Критерий: минимум эмпирического риска

$$Q(w) = \sum_{i=1}^{\ell} \mathscr{L}(a(x_i, w), y_i) + \mathscr{R}(w) \to \min_w,$$

$\mathscr{L}(a, y)$ — функция потерь (отличие a от правильного ответа y)
$\mathscr{R}(w)$ — регуляризатор, дополнительные требования к модели

Основные типы задач обучения с учителем:

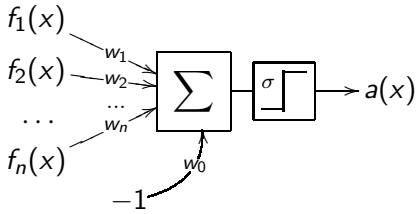
- регрессия: $y_i \in \mathbb{R}$, $\mathscr{L}(a, y) = (a - y)^2$
- классификация: $y_i \in \{-1, +1\}$, $\mathscr{L}(a, y) = [\text{sign}\, a \neq y] = [ay < 0] \leqslant L(ay)$

Искусственный нейрон — линейная модель классификации

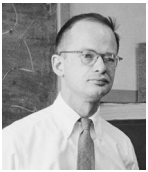
Линейная модель нейрона (1943):

$$a(x, w) = \sigma\left(\sum_{j=1}^{n} w_j f_j(x) - w_0\right)$$

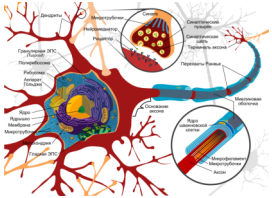
$f_j(x)$ — признаки объекта x
w_j — веса признаков, w_0 — порог активации
$\sigma(z)$ — функция активации $\operatorname{sign} z$, $\operatorname{th} z$, $\frac{1}{1+e^{-z}}, \ldots$



Уоррен МакКаллок (1898–1969)

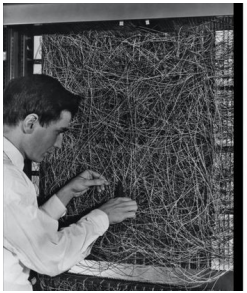


Вальтер Питтс (1923–1969)



Персептрон Розенблатта (1957) и теорема Новикова (1960)

Mark-1 — первый нейрокомпьютер (1960)
Обучение — метод коррекции ошибки
Архитектура — двухслойная нейронная сеть



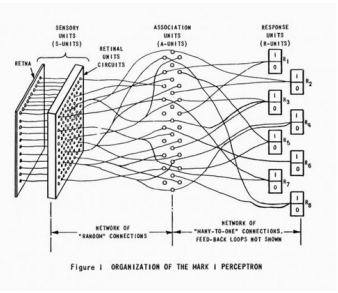
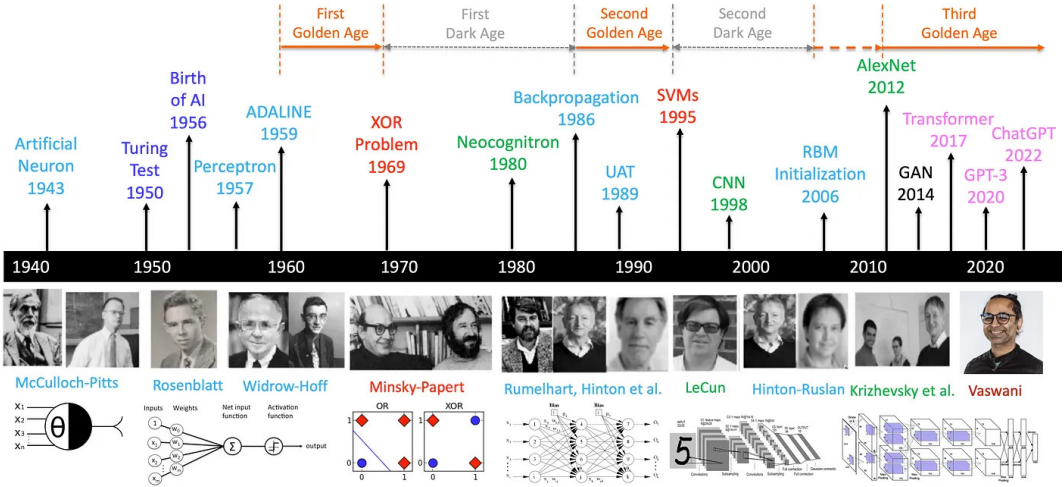


Figure 1 ORGANIZATION OF THE MARK I PERCEPTRON

Фрэнк Розенблатт (1928–1971)

Розенблатт Ф. Принципы нейродинамики. Перцептроны и теория механизмов мозга. 1965 (1962).
Novikoff A. B. J. On convergence proofs on perceptrons. 1962.

Основные вехи развития нейронных сетей



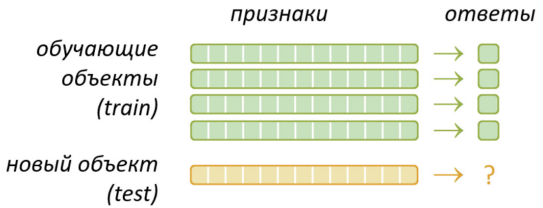
Джеффри Хинтон и основные вехи развития теории нейронных сетей

рекуррентные сети	*Hopfield J.* Neural networks and physical systems with emergent collective computational abilities. 1982. *Ackley D.H., Hinton G., Sejnowski T.J.* A learning algorithm for Boltzmann machines. 1985. *Hochreiter S., Schmidhuber J.* Long short-term memory. 1997.
метод BackProp	*Галушкин Александр Иванович.* Синтез многослойных систем распознавания образов. 1974. *Rummelhart D., Hinton G., Williams R.* Learning internal representations by error propagation. 1985. *LeCun Y.* Une procedure d'aprentissage pour reseau a seuil assymetrique. 1985. *Parker D. B.* Learning-logic: Casting the cortex of the human brain in silicon. 1985.
глубокие сети	*Ивахненко А. Г., Лапа В. Г.* Кибернетические предсказывающие устройства. 1965. *Rina Dechter.* Learning while searching in constraint-satisfaction problems. 1986. *Hinton G.* Learning multiple layers of representation. 2007.
свёрточные сети	*LeCun, Bottou, Bengio, Haffner.* Gradient-based learning applied to document recognition. 1998. *Krizhevsky, Sutskever, Hinton G.* ImageNet classification with deep convolutional neural networks. 2012.
капсульные сети	*Sabour S., Frosst N., Hinton G.* Dynamic Routing Between Capsules. 2017.
генеративные сети	*Hinton G., Osindero S., Teh Y.W.* A fast learning algorithm for deep belief nets. 2006. *Kingma D.P., Welling M.* Auto-Encoding Variational Bayes. 2013. *Goodfellow I. et al.* Generative Adversarial Nets. 2014. *Vaswani A. et al.* Attention is all you need. 2017.

Три основных этапа. Этап 1: вектор → скаляр

Предсказательное моделирование векторных данных

Вход: векторные признаковые описания объектов
Выход: скалярные ответы (решения, классификации, предсказания, прогнозы)



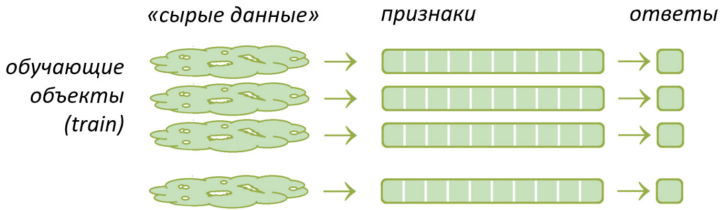
Приложения: медицинская диагностика, геологическое прогнозирование, кредитный скоринг, прогнозирование объёмов перевозок, продаж,...

Модели: kNN, NB, SVM, RBF, LR, MVR, GLM, MLP, ID3, CART, RF, GBM,...

Три основных этапа. Этап 2: структура → вектор → скаляр

Обучаемая векторизация сложно структурированных данных

Вход: сложно структурированные «сырые» данные объектов
Выход: векторные представления объектов, затем ответы



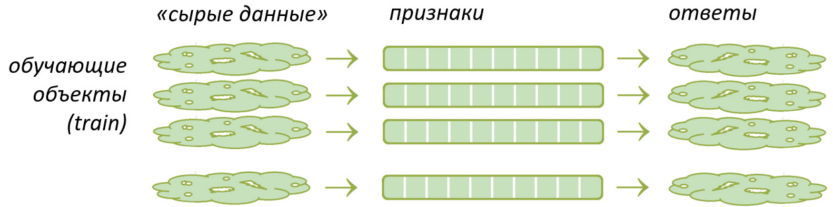
Приложения: классификация изображений, текстов, сигналов, голосовых команд, информационный поиск и рекомендации, биометрическая идентификация,...

Модели: CNN, AlexNet, ResNet, SNE, tSNE, GNN, word2vec, FastText, BERT,...

Три основных этапа. Этап 3: структура → вектор → структура

Обучаемая генерация сложно структурированных данных

Вход: сложно структурированные объекты
Выход: сложно структурированные ответы

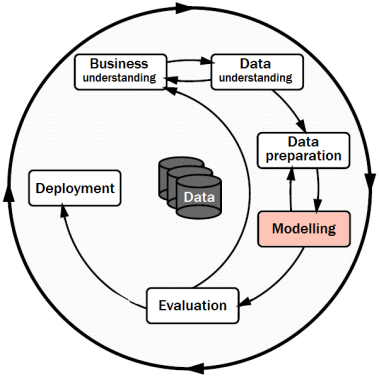


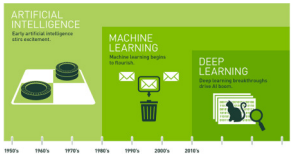
Приложения: синтез изображений и видео, перенос стиля, распознавание речи, машинный перевод, суммаризация текстов, чат-боты,...

Модели: seq2seq, RNN, LSTM, GAN, VAE, DALL-E, GPT, LLM,...

Понимание эволюции ИИ как автоматизации шагов CRISP-DM

CRISP-DM: CRoss Industry Standard Process for Data Mining (1999)

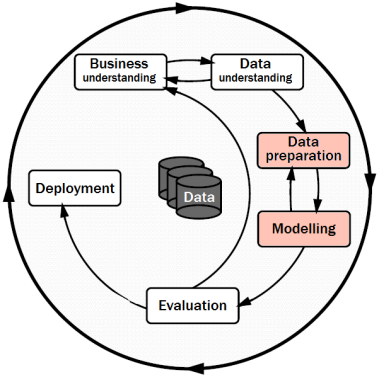


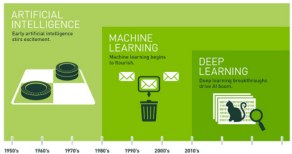


- *Expert Systems:*
  жёсткие модели, основанные на правилах
- *Machine Learning:*
  параметрические модели, обучаемые по данным

Понимание эволюции ИИ как автоматизации шагов CRISP-DM

CRISP-DM: CRoss Industry Standard Process for Data Mining (1999)

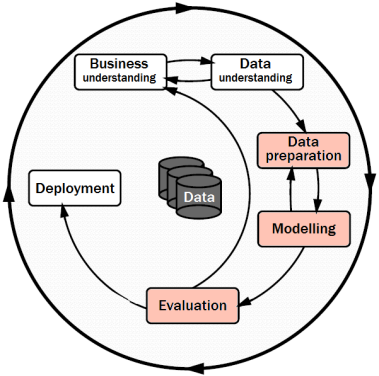


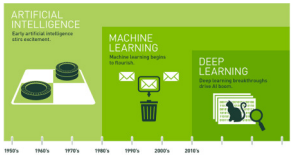


- *Expert Systems:*
  жёсткие модели, основанные на правилах
- *Machine Learning:*
  параметрические модели, обучаемые по данным
- *Deep Learning:*
  модели с обучаемой векторизацией данных

Понимание эволюции ИИ как автоматизации шагов CRISP-DM

CRISP-DM: CRoss Industry Standard Process for Data Mining (1999)

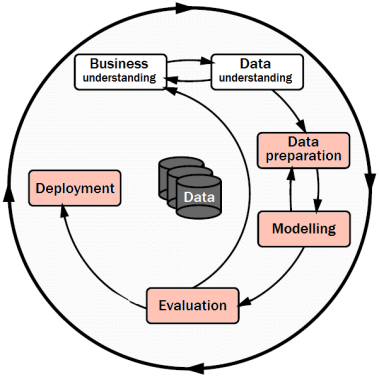


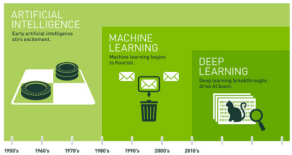


- *Expert Systems:*
  жёсткие модели, основанные на правилах
- *Machine Learning:*
  параметрические модели, обучаемые по данным
- *Deep Learning:*
  модели с обучаемой векторизацией данных
- *AutoML:*
  автоматический выбор моделей и архитектур

Понимание эволюции ИИ как автоматизации шагов CRISP-DM

CRISP-DM: CRoss Industry Standard Process for Data Mining (1999)





- *Expert Systems:*
  жёсткие модели, основанные на правилах
- *Machine Learning:*
  параметрические модели, обучаемые по данным
- *Deep Learning:*
  модели с обучаемой векторизацией данных
- *AutoML:*
  автоматический выбор моделей и архитектур
- *Lifelong Learning:*
  бесшовная интеграция моделей в бизнес-процесс

Сеть Хопфилда: постановка задачи

Дано: обучающая выборка объектов $X^\ell = \{x_1, \ldots, x_\ell\} \subset \{-1, +1\}^n$

Найти: модель ассоциативной памяти $a(x, w)$, способную выдавать объект x_i из X^ℓ, ближайший к входному вектору x

Критерий: минимум «энергии»

$$E(x) = -\frac{1}{2}\sum_{i=1}^{\ell}\langle x_i, x\rangle^2 = -\frac{1}{2}x^{\mathsf{T}}\left(\sum_{i=1}^{\ell} x_i x_i^{\mathsf{T}}\right)x = -\frac{1}{2}x^{\mathsf{T}}Wx \to \min_x$$

что эквивалентно поиску наиболее вероятного состояния x
с минимальной энергией в модели Изинга без внешнего поля

$$p(x|W) = \frac{1}{Z(W)}\exp\left(-\frac{1}{2}x^{\mathsf{T}}Wx - \cancel{x^{\mathsf{T}}a}\right) \to \max_x$$

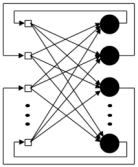
Hopfield J. Neural networks and physical systems with emergent collective computational abilities. 1982.

Сеть Хопфилда: «архитектура сети» и метод обучения сети

Метод внешнего произведения (Хопфилд):

$$W = \frac{1}{\ell}\sum_{i=1}^{\ell} x_i x_i^{\mathrm{T}} - I_n$$

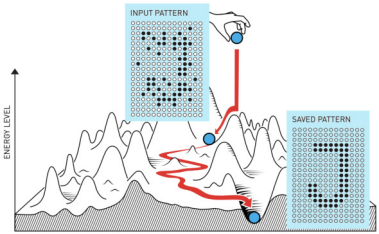
W — $n \times n$-матрица, $w_{jk} = w_{kj}$, $w_{jj} = 0$



Модель ассоциативной памяти a: $x^{(0)} \mapsto x$ реализуется итерационным процессом

$$x^{(t+1)} = \operatorname{sign}\bigl(Wx^{(t)}\bigr),$$

сходится к одному из образов x_i, если $\ell \lessapprox \frac{n}{2\ln n}$



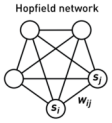
Hopfield J. Neural networks and physical systems with emergent collective computational abilities. 1982.

От машин Больцмана к глубоким сетям доверия

Критерий: минимум энергии (Boltzmann Machine)

$$E(x) = -\frac{1}{2}x^{\mathrm{T}}Wx - x^{\mathrm{T}}a \to \min_{x}$$

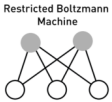
вводит в модель внешнее поле и параметр температуры,
использует метод имитации отжига для улучшения сходимости.



Критерий: минимум энергии (Restricted Boltzmann Machine)

$$E(x) = -\frac{1}{2}x^{\mathrm{T}}Wh - x^{\mathrm{T}}a - h^{\mathrm{T}}b \to \min_{x,h}$$

обогащает модель, вводя вектор скрытых переменных h,
позволяет использовать вещественные значения.



Ackley D.H., *Hinton G.E.*, *Sejnowski T.J.* A learning algorithm for Boltzmann machines. 1985.
Hinton G.E., *Osindero S.*, *Teh Y.W.* A fast learning algorithm for deep belief nets. 2006.
Hinton G.E., *Salakhutdinov R.* A better way to pretrain deep Boltzmann machines. 2012.

Построение автокодировщика — задача обучения без учителя

Дано: $X^\ell = \{x_1, \ldots, x_\ell\}$ — обучающая выборка
Найти: $f: X \to Z$ — кодировщик (encoder), преобразует x в кодовый вектор $z = f(x, \alpha)$
$g: Z \to X$ — декодировщик (decoder), преобразует z в реконструкцию $\hat{x} = g(z, \beta)$

Критерий: Суперпозиция $\hat{x} = g(f(x))$ должна восстанавливать исходные x_i:

$$Q_{\text{AE}}(\alpha, \beta) = \sum_{i=1}^{\ell} \mathscr{L}\big(g(f(x_i, \alpha), \beta), x_i\big) \to \min_{\alpha,\beta}$$

Квадратичная функция потерь: $\mathscr{L}(\hat{x}, x) = \|\hat{x} - x\|^2$

Пример 1. Линейный автокодировщик: $x \in \mathbb{R}^n$, $z \in \mathbb{R}^m$

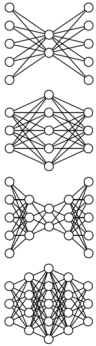
$$f(x, A) = \underset{m\times n}{A}\, x, \qquad g(z, B) = \underset{n\times m}{B}\, z$$

Пример 2. Двухслойная сеть с функциями активации σ_f, σ_g:

$$f(x, A) = \sigma_f(Ax + a), \qquad g(z, B) = \sigma_g(Bz + b)$$

Способы использования автокодировщиков

- Генерация признаков (feature generation)
- Снижение размерности (dimensionality reduction)
- Сжатие данных с минимальными потерями точности
- Повышение информативности признакового пространства при решении задач обучения с учителем
- Обучаемая векторизация объектов, встраиваемая в архитектуру глубокой нейронной сети
- Послойное предобучение многослойных сетей
- Генерация синтетических объектов, похожих на реальные



D.Rumelhart, *G.Hinton*, *R.Williams*. Learning internal representations by error propagation, 1985.

David Charte et al. A practical tutorial on autoencoders for nonlinear feature fusion: taxonomy, models, software and guidelines. 2018.

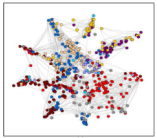
Векторизация объектов по данным о расстояниях (Multidimensional Scaling)

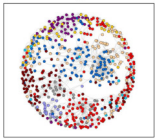
Дано: объекты — вершины графа, $(i,j) \in E$ — выборка рёбер графа $\langle V, E\rangle$
R_{ij} — расстояния между вершинами ребра (i,j)

Найти: векторные представления вершин $z_i \in \mathbb{R}^d$ заданной размерности d

Критерий: близкие (по графу) вершины должны иметь близкие векторы

$$Q(Z) = \sum_{(i,j)\in E} \big(\|z_i - z_j\| - R_{ij}\big)^2 \to \min_Z$$





- Это задача *многомерного шкалирования*,
- способ визуализации кластерных структур (при $d = 2$)
- **Недостаток:** при проецировании неизбежны искажения,
- особенно при локальных перепадах плотности точек

G.E.Hinton, S.T.Roweis. Stochastic Neighbor Embedding. 2002.
Laurens van der Maaten, Geoffrey Hinton. Visualizing data using t-SNE. 2008

Векторизация по отношению соседства (Stochastic Neighbor Embedding, t-SNE)

Дано: объекты — вершины графа, $(i,j) \in E$ — выборка рёбер графа $\langle V, E \rangle$
R_{ij} — расстояния между вершинами ребра (i,j)

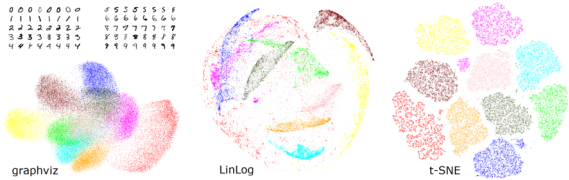
Найти: векторные представления вершин $z_i \in \mathbb{R}^d$ заданной размерности d

Критерий: распределения $P(j$ является соседом $i)$ должны быть близки:

$$Q(z) = \sum_{j,i} p(j,i) \ln q(j,i) \;\to\; \max_z$$

в исходном пространстве:
$p(j,i) \propto \exp\left(-\frac{1}{2\sigma_i^2} R_{ij}^2\right)$

в пространстве проекции:
$q(j,i) \propto \exp\left(-\|z_i - z_j\|^2\right)$



G.E.Hinton, S.T.Roweis. Stochastic Neighbor Embedding. 2002.
Laurens van der Maaten, Geoffrey Hinton. Visualizing data using t-SNE. 2008

Вариационный автокодировщик (Variational AE) для генерации объектов

Дано: $X^\ell = \{x_1, \ldots, x_\ell\}$ — обучающая выборка

Найти: $q_\alpha(z|x)$ — вероятностный кодировщик с параметром α
$p_\beta(\hat{x}|z)$ — вероятностный декодировщик с параметром β

Критерий: Максимизация нижней оценки log-правдоподобия:

$$Q_{\text{VAE}}(\alpha,\beta) = \sum_{i=1}^{\ell} \log p(x_i) = \sum_{i=1}^{\ell} \log \int q_\alpha(z|x_i) \frac{p_\beta(x_i|z)p(z)}{q_\alpha(z|x_i)} dz \geqslant$$

$$\geqslant \sum_{i=1}^{\ell} \int q_\alpha(z|x_i) \log \frac{p_\beta(x_i|z)p(z)}{q_\alpha(z|x_i)} dz =$$

$$= \sum_{i=1}^{\ell} \int q_\alpha(z|x_i) \log p_\beta(x_i|z) dz - \text{KL}\big(q_\alpha(z|x_i) \,\big\|\, p(z)\big) \to \max_{\alpha,\beta}$$

D.P.Kingma, M.Welling. Auto-encoding Variational Bayes. 2013.
C.Doersch. Tutorial on variational autoencoders. 2016.

Вариационный автокодировщик (Variational AE) для генерации объектов

Оптимизационная задача для вариационного автокодировщика:

$$Q_{\text{VAE}}(\alpha,\beta) = \sum_{i=1}^{\ell} \underbrace{\mathsf{E}_{z\sim q_\alpha(z|x_i)} \log p_\beta(x_i|z)}_{\substack{\text{качество реконструкции} \\ \approx \log p_\beta(x_i|z),\ z\sim q_\alpha(z|x_i)}} - \underbrace{\mathsf{KL}\big(q_\alpha(z|x_i) \,\big\|\, p(z)\big)}_{\text{регуляризатор по } \alpha} \to \max_{\alpha,\beta}$$

где $p(z)$ — априорное распределение, обычно $\mathcal{N}(0,\sigma^2 I)$

Репараметризация $q_\alpha(z|x_i)$: $\quad z = f(x_i,\alpha,\varepsilon),\ \varepsilon\sim\mathcal{N}(0,I)$

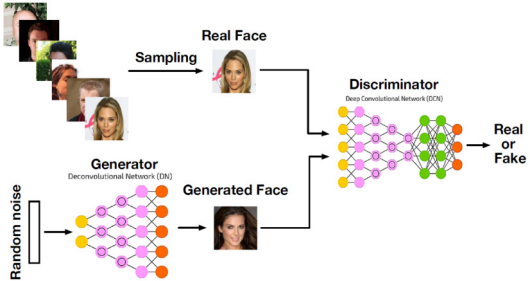
Метод стохастического градиента:

- сэмплировать $x_i \sim X^\ell,\ \ \varepsilon\sim\mathcal{N}(0,I),\ \ z = f(x_i,\alpha,\varepsilon)$
- $\alpha := \alpha + h\nabla_\alpha\big[\log p_\beta\big(x_i|f(x_i,\alpha,\varepsilon)\big) - \mathsf{KL}\big(q_\alpha(z|x_i)\|p(z)\big)\big]$
  $\beta := \beta + h\nabla_\beta\big[\log p_\beta(x_i|z)\big]$

Генерация объектов, похожих на исходные: $x\sim p_\beta\big(x|f(x_i,\alpha,\varepsilon)\big),\ \ \varepsilon\sim\mathcal{N}(0,I)$

Генеративная состязательная сеть (Generative Adversarial Net)

Генератор $G(z)$ учится порождать объекты x из шума z
Дискриминатор $D(x)$ учится отличать их от реальных объектов



Antonia Creswell et al. Generative Adversarial Networks: an overview. 2017.
Zhengwei Wang, *Qi She*, *Tomas Ward.* Generative Adversarial Networks: a survey and taxonomy. 2019.
Chris Nicholson. A Beginner's Guide to Generative Adversarial Networks.
https://pathmind.com/wiki/generative-adversarial-network-gan. 2019.

Постановка задачи GAN

Дано: выборка объектов $\{x_i\}_{i=1}^{\ell}$ из X

Найти: вероятностную генеративную модель $G(z,\alpha)$: $x \sim p(x|z,\alpha)$
вероятностную дискриминативную модель $D(x,\beta) = p(1|x,\beta)$

Критерий: модели играют друг с другом в антагонистическую игру

Обучение дискриминативной модели D по максимуму правдоподобия:

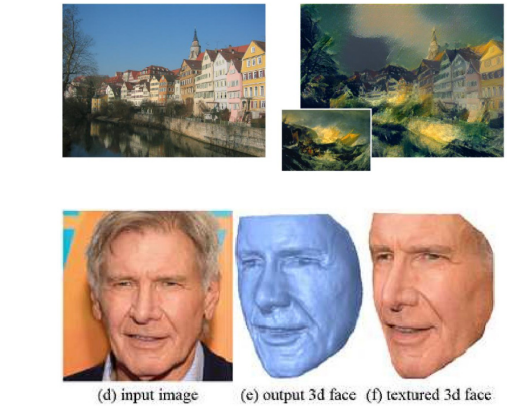
$$\sum_{i=1}^{\ell} \ln D(x_i,\beta) + \ln\big(1 - D(G(z_i,\alpha),\beta)\big) \;\to\; \max_{\beta}$$

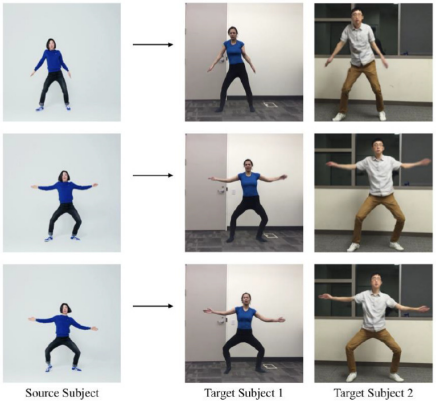
Обучение генеративной модели G по случайному шуму $\{z_i\}_{i=1}^{m}$:

$$\sum_{i=1}^{\ell} \ln\big(1 - D(G(z_i,\alpha),\beta)\big) \;\to\; \min_{\alpha}$$

Ian Goodfellow et al. Generative Adversarial Nets. 2014

Примеры GAN для синтеза изображений и видео





Chuan Li, M.Wand. Precomputed real-time texture synthesis with Markovian generative adversarial networks. 2016.
Xiaoxing Zeng, Xiaojiang Peng, Yu Qiao. DF2Net: a dense fine finer network for detailed 3D face reconstruction. 2019.
Caroline Chan, Shiry Ginosar, Tinghui Zhou, Alexei A. Efros. Everybody dance now. ICCV-2019.

Полносвязная нейронная сеть с L слоями

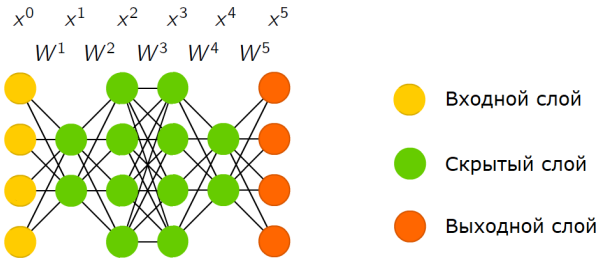
Архитектура сети: H_l — число нейронов в l-м слое, $l = 1, \ldots, L$

$x^0 = x = \big(f_j(x)\big)_{j=0}^{n}$ — вектор признаков на входе сети, $H_0 = n$

$x^l = \big(x_h^l\big)_{h=0}^{H_l}$ — вектор признаков на выходе l-го слоя, $x_0^l = -1$

$x^L = a(x) = \big(a_m(x)\big)_{m=1}^{M}$ — выходной вектор сети, $H_L = M$

$W^l = (w_{kh}^l)$ — матрица весов l-го слоя, размера $(H_{l-1}+1) \times H_l$



Метод стохастического градиента SG (Stochastic Gradient)

Дано: обучающая выборка объектов, $\{x_i\}_{i=1}^{\ell}$, $x_i \in \mathbb{R}^n$

Найти: матрицы весов всех слоёв глубокой сети $w = (W^1, \ldots, W^L)$

Критерий: минимум потерь на выборке: $Q(w) = \frac{1}{\ell} \sum_{i=1}^{\ell} \mathscr{L}_i(w) \to \min_w$

Вход: выборка $(x_i, y_i)_{i=1}^{\ell}$; темп обучения η; параметр λ;

Выход: вектор весов всех слоёв $w = (W^1, \ldots, W^L)$;

инициализировать веса w и текущую оценку $Q(w)$;

повторять

- выбрать объект x_i из X^ℓ (например, случайно);
- градиентный шаг: $w := w - \eta \nabla \mathscr{L}_i(w)$;
- вычислить потерю $\mathscr{L}_i(w)$;
- оценить функционал скользящим средним: $Q := (1 - \lambda) Q + \lambda \mathscr{L}_i(w)$;

пока значение Q и/или веса w не стабилизируются;

Задача дифференцирования суперпозиции функций

Вычисление сети по входному вектору x, рекуррентно по слоям:

$$x_h^l = \sigma_h^l(S_h^l), \quad S_h^l = \sum_{k=0}^{H_{l-1}} w_{kh}^l x_k^{l-1}, \quad h = 1, \ldots, H_l, \quad l = 1, \ldots, L,$$

то же самое в матричной записи: $x^l = \sigma^l\big(W^l x^{l-1}\big)$.

Функция потерь на объекте x_i (квадратичная для регрессии с m-мерным выходом):

$$\mathscr{L}_i(w) = \frac{1}{2} \sum_{m=1}^{M} \big(a_m(x_i, w) - y_{im}\big)^2$$

По формуле дифференцирования суперпозиции функций:

$$\frac{\partial \mathscr{L}_i(w)}{\partial w_{kh}^l} = \frac{\partial \mathscr{L}_i(w)}{\partial x_h^l} \frac{\partial x_h^l}{\partial w_{kh}^l}, \quad k = 0, \ldots, H_{l-1}, \quad h = 1, \ldots, H_l$$

D.Rumelhart, *G.Hinton*, *R.Williams*. Learning internal representations by error propagation, 1985.

Рекуррентное вычисление частных производных

Найдём сначала частные производные $\mathscr{L}_i(w)$ по $x_m^L \equiv a_m(x_i, w)$:

$$\frac{\partial \mathscr{L}_i(w)}{\partial x_m^L} = a_m(x_i, w) - y_{im} \equiv \varepsilon_{im}^L;$$

для квадратичной функции потерь это *ошибка на m-м нейроне выходного слоя*.

Частные производные по x_h^l вычислим по уровням справа налево, $l = L, \ldots, 2$:

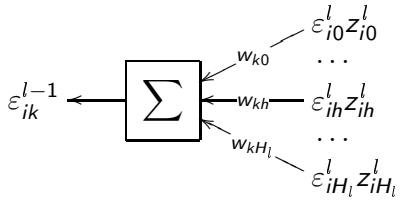
$$\frac{\partial \mathscr{L}_i(w)}{\partial x_k^{l-1}} = \sum_{h=0}^{H_l} \frac{\partial \mathscr{L}_i(w)}{\partial x_h^l} \underbrace{(\sigma_h^l)'(S_{ih}^l)}_{z_{ih}^l} w_{kh}^l = \sum_{h=0}^{H_l} \varepsilon_{ih}^l z_{ih}^l w_{kh}^l = \varepsilon_{ik}^{l-1}$$

— формально назовём это *ошибкой на k-м нейроне $(l-1)$-го скрытого слоя*.

Замечание: функция активации σ_h^l и её производная $(\sigma_h^l)'$ вычисляются в одной и той же точке $S_{ih}^l = \sum_{k=0}^{H_{l-1}} w_{kh}^l x_{ik}^{l-1}$

Быстрое вычисление градиента

Рекуррентная формула записана так, будто сеть запускается «задом наперёд», чтобы вычислять ε_{ik}^{l-1} по ε_{ih}^{l}:



Имея частные производные $\mathscr{L}_i(w)$ по x_h^l, находим компоненты градиента $\nabla\mathscr{L}$:

$$\frac{\partial \mathscr{L}_i(w)}{\partial w_{kh}^l} = \frac{\partial \mathscr{L}_i(w)}{\partial x_h^l} \frac{\partial x_h^l}{\partial w_{kh}^l} = \varepsilon_{ih}^l z_{ih}^l x_{ik}^{l-1}$$

Алгоритм обратного распространения ошибки BackProp

Вход: выборка $(x_i, y_i)_{i=1}^{\ell}$, архитектура $(H_l)_{l=1}^{L}$, параметры η, λ, инициализация $w := w_0$;
Выход: вектор весов всех слоёв $w = (W^1, \ldots, W^L)$;
повторять
  выбрать объект x_i из X^ℓ (например, случайно);
  для всех $l = 1, \ldots, L$, $h = 1, \ldots, H_l$ прямой ход:
    $S_{ih}^l := \sum_{k=0}^{H_{l-1}} w_{kh}^l x_{ik}^{l-1}$; $x_{ih}^l := \sigma_h^l(S_{ih}^l)$; $z_{ih}^l := (\sigma_h^l)'(S_{ih}^l)$;
  для всех $h = 1, \ldots, H_L$ расчёт ошибок:
    $\varepsilon_{hi}^L := \frac{\partial \mathscr{L}_i(w)}{\partial x_h^L}$;
  для всех $l = L, \ldots, 2$, $k = 0, \ldots, H_{l-1}$ обратный ход:
    $\varepsilon_{ik}^{l-1} := \sum_{h=0}^{H_l} \varepsilon_{ih}^l z_{ih}^l w_{kh}^l$;
  для всех $l = 1, \ldots, L$, $k = 0, \ldots, H_{l-1}$, $h = 1, \ldots, H_l$ градиентный шаг:
    $w_{kh}^l := w_{kh}^l - \eta\, \varepsilon_{ih}^l z_{ih}^l x_{ik}^{l-1}$;
пока значения Q и/или веса w не стабилизируются;

Алгоритм BackProp: преимущества и недостатки

Преимущества:

- время вычисления градиента $O(\dim w)$ вместо $O(\dim^2 w)$
- обобщение на любые σ, $\mathscr{L}$, любые архитектуры
- возможность динамического (потокового) обучения
- возможно сублинейное обучение на больших выборках (когда части объектов x_i уже достаточно для обучения)
- возможно распараллеливание

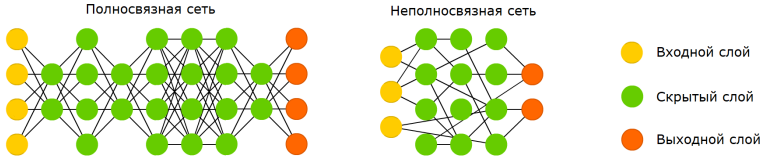
Недостатки:

- медленная сходимость, застревание в локальных экстремумах
- возможны «затухание» и «взрывы» градиентов
- возможно переобучение
- подбор комплекса эвристик является искусством

D.Rumelhart, *G.Hinton*, *R.Williams*. Learning internal representations by error propagation, 1985.

Глубокие нейронные сети (Deep Neural Network, DNN)

1965: первые глубокие нейронные сети
2012: свёрточная сеть для классификации изображений AlexNet



- *Архитектура сети* — структура слоёв и связей между ними, позволяющая наделять DNN нужными свойствами
- DNN позволяют принимать на входе и генерировать на выходе *сложно структурированные данные*

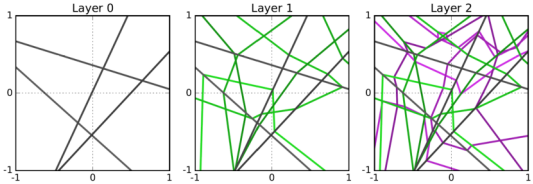
Ива́хненко А. Г., Лапа В. Г. Кибернетические предсказывающие устройства. 1965
Krizhevsky A., Sutskever I., Hinton G. ImageNet classification with deep convolutional neural networks. 2012.

Глубина важнее ширины

A^n_{LH} — семейство полносвязных многослойных сетей $a(x, w)$: n признаков, L слоёв, H нейронов в каждом слое, $x \in \mathbb{R}^n$, функции активации кусочно-линейные (ReLU)

Мера разнообразия семейства A^n_{LH} — максимальное число участков линейности $a(x, w)$ — выпуклых многогранников в $\mathbb{R}^n$.

Пример. Участки линейности, $n = 2$, $L = 3$, $H = 4$:



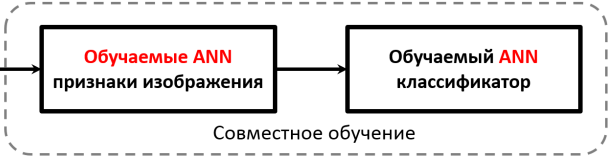
Теорема. Разнообразие семейства A^n_{LH} растёт как $O(H^{nL})$.

M.Raghu et al. On the Expressive Power of Deep Neural Networks, 2016.

Генерация признаков для распознавания изображений

Классический подход к распознаванию изображений:

Современный подход — end-to-end deep learning:



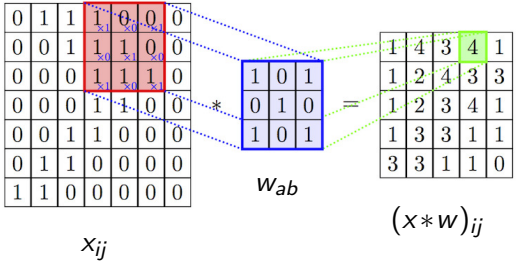
Sanjeev Arora. Toward theoretical understanding of deep learning. ICML-2018 Tutorial
https://unsupervised.cs.princeton.edu/deeplearningtutorial.html

Свёрточный слой нейронов (convolution layer)

x_{ij} — исходные признаки, пиксели $n \times m$-изображения
w_{ab} — ядро свёртки, $a = -A, \ldots, +A$, $b = -B, \ldots, +B$

Неполносвязный свёрточный нейрон с $(2A+1)(2B+1)$ весами:

$$(x*w)_{ij} = \sum_{a=-A}^{A} \sum_{b=-B}^{B} w_{ab}\, x_{i+a,j+b}$$



Yann LeCun et al. Learning algorithms for classification: A comparison on handwritten digit recognition. 1995

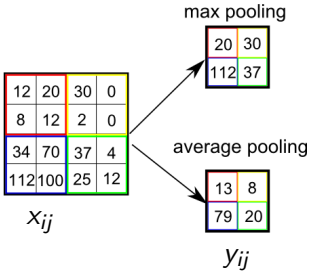
Объединяющий слой нейронов (pooling layer)

Объединяющий нейрон — это необучаемая свёртка с шагом $h > 1$, агрегирующая данные прямоугольной области размера $h \times h$:

$$y_{ij} = F\left(x_{hi,hj}, \ldots, x_{hi+h-1,\, hj+h-1}\right),$$

где F — агрегирующая функция: max, average и т.п.

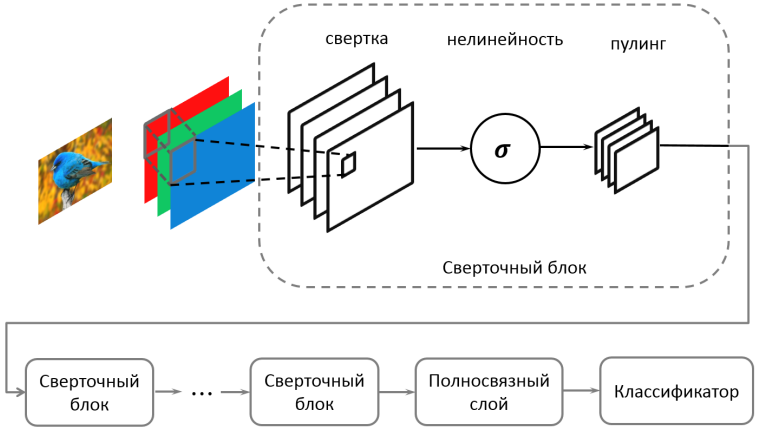
Размер изображения при переходе $y \to x$ сокращается в h раз по ширине и по высоте



если нейрон предыдущего слова отвечал за детектирование некоторого элемента,
то max-pooling позволяет обнаружить этот элемент в любом месте из h-окрестности (инвариантность детектирования относительно сдвигов)

Yann LeCun et al. Learning algorithms for classification: A comparison on handwritten digit recognition. 1995

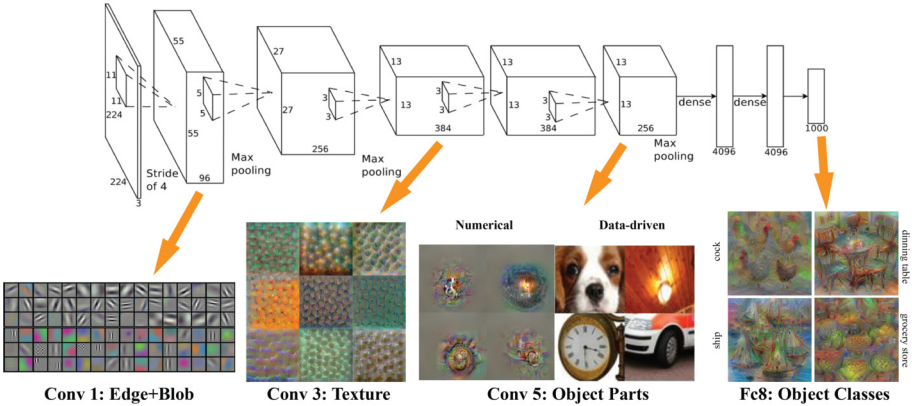
Стандартная схема сверточной сети (Convolutional NN)



Yann LeCun et al. Learning algorithms for classification: A comparison on handwritten digit recognition. 1995

Свёрточная сеть для распознавания объектов на изображениях ImageNet

Каждый слой распознаёт всё более крупные и сложные элементы изображения



Krizhevsky A., Sutskever I., Hinton G. ImageNet classification with deep convolutional neural networks. 2012.

ImageNet — большая выборка размеченных изображений

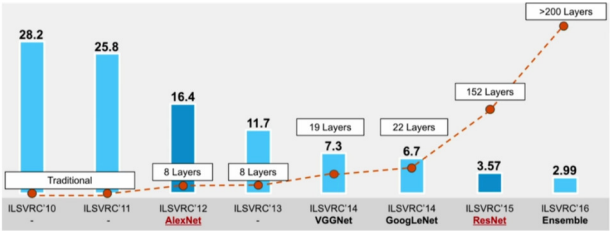
https://image-net.org
2,5 года на разметку
(2008/07–2010/04)

14 197 122 изображений
21 841 классов объектов

3 разметки каждого
изображения

Li Fei-Fei et al. ImageNet: A large-scale hierarchical image database. 2009.
Li Fei-Fei et al. Construction and analysis of a large scale image ontology. 2009.

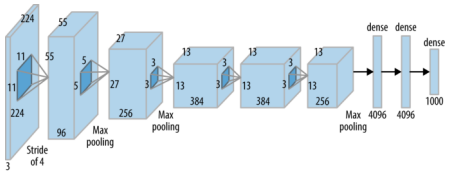
Глубокие свёрточные сети для классификации изображений



Старт в 2009. Человеческий уровень ошибок 5% пройден в 2015

Свёрточная сеть **AlexNet**:

+ ImageNet + 60M параметров + GPU
+ ReLU + Dropout
+ augmentation (пополнение) выборки
+ подбор размеров слоёв и свёрток



Krizhevsky A., Sutskever I., Hinton G. ImageNet classification with deep convolutional neural networks. 2012.

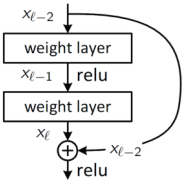
Сеть со сквозными связями ResNet (Residual Neural Network)

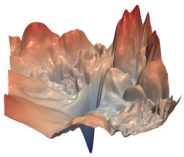
Сквозная связь (skip connection)
слоя ℓ с предшествующим слоем $\ell - d$:

$$x_\ell = \sigma(W x_{\ell-1}) + x_{\ell-d}$$

Регуляризирующее воздействие:
слой ℓ строит не новый вектор x_ℓ,
а малое приращение $x_\ell - x_{\ell-d}$

Нетривиальный результат:
упрощается ландшафт оптимизируемого критерия,
устраняются локальные экстремумы и седловые точки




без сквозных связей

со сквозными связями

Kaiming He et al. Deep residual learning for image recognition. 2015
Hao Li et al. Visualizing the loss landscape of neural nets. 2018

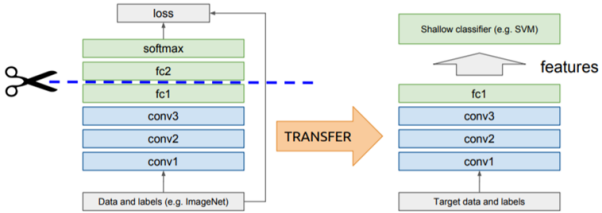
Предобучение (pre-training), перенос обучения (transfer learning)

Обучение модели векторизации $z = f(x, \alpha)$ на выборке $\{x_i\}_{i=1}^{\ell}$:

$$\sum_{i=1}^{\ell} \mathscr{L}_i\big(g(f(x_i, \alpha), \beta)\big) \;\to\; \min_{\alpha,\beta}$$

Обучение целевой модели $y = g(z, \beta)$ на другой выборке $\{x'_i\}_{i=1}^{m}$, как правило, малого объёма:

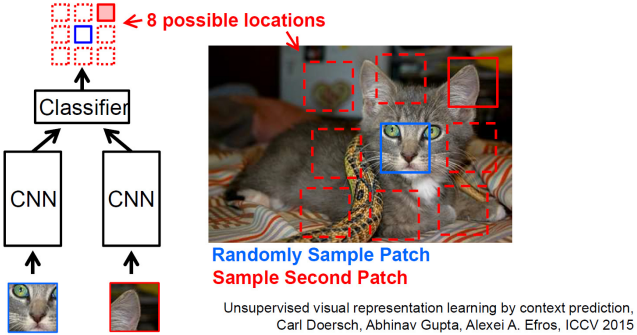
$$\sum_{i=1}^{m} \mathscr{L}'_i\big(g'(f(x'_i, \alpha), \beta')\big) \;\to\; \min_{\beta'}$$



Sinno Jialin Pan, Qiang Yang. A Survey on Transfer Learning. 2009
Jason Yosinski, Jeff Clune, Yoshua Bengio, Hod Lipson. How transferable are features in deep neural networks? 2014

Самостоятельное обучение (self-supervised learning)

Модель векторизации $z = f(x, \alpha)$ обучается предсказывать взаимное расположение пар фрагментов одного изображения

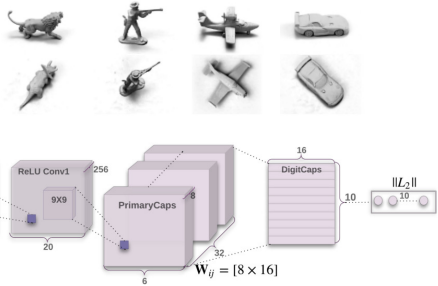


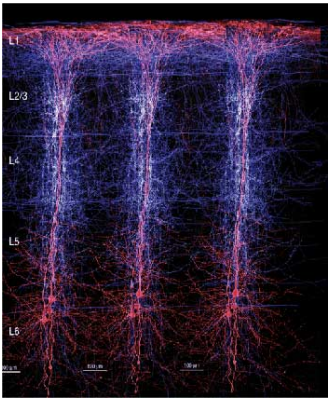
Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

Преимущество: сеть выучивает векторные представления объектов без размеченной обучающей выборки (без ImageNet).

Капсульная нейронная сеть Хинтона

Мотивация: микроколонки в коре мозга способны узнавать объекты независимо от расположения, освещения. Однако max-pooling на это не способен.





Sabour S., *Frosst N.*, *Hinton G.E.* Dynamic Routing Between Capsules. 2017.
Hinton G.E., *Sabour S.*, *Frosst N.* Matrix capsules with EM routing. 2018.

Сухой остаток

Три этапа развития технологий машинного обучения:

1. вектор $\rightarrow$ скаляр
2. структура $\rightarrow$ вектор $\rightarrow$ скаляр
3. структура $\rightarrow$ вектор $\rightarrow$ структура

Достижения нобелевских лауреатов Джона Хопфилда и Джеффри Хинтона, способствовали переходу к этапам 2 и 3:

- сети ассоциативной памяти
- метод обратного распространения ошибок
- обучаемая векторизация данных, сеть AlexNet
- свёрточные и капсульные нейронные сети
- автокодировщики и генеративные сети

Полезные ссылки и литература

- *Кевин Мэрфи.* Вероятностное машинное обучение. Введение. 2022.
- *Визильтер Ю. В.*
  От слабого ИИ к общему универсальному интеллекту (обзор тенденций 2020-2023).
  Семинар РАИИ и ФИЦ ИУ РАН «Проблемы искусственного интеллекта» 31-01-2024
  https://rutube.ru/video/2aad53ec833f19918c1593398a2a1b88
- Не пропустите открытие тысячелетия! Vital Math, 13-01-2024,
  https://www.youtube.com/watch?v=JZjH0it9Jyg
- Report: AI Decrypted: A Guide for Navigating AI Developments in 2024, Навигатор по ИИ-ландшафту от Dentons Global Advisors, 24-01-2024
  https://www.albrightstonebridge.com/news/report-ai-decrypted-guide-navigating-ai-developments-2024
- *Воронцов К. В.* Лекции по машинному обучению. МФТИ, МГУ.
  www.MachineLearning.ru, User:Vokov, 2004–2024. https://bit.ly/ML-Vorontsov
- *Гарбук С. В., Губинский А. М.* Искусственный интеллект в ведущих странах мира: стратегии развития и военное применение. 2020.
- *Шумский С. А.* Машинный интеллект. РИОР ИНФРА-М, 2020.