

Автокодировщики: введение и примеры использования

Рита Кузнецова
Московский физико-технический институт
Компания Антиплагиат

17 ноября 2017

Автокодировщик

- ▶ Автокодировщик — это модель снижения размерности

$$\mathbf{X} \xrightarrow[\text{encode}]{f} \mathbf{H} \xrightarrow[\text{decode}]{g} \mathbf{X},$$

$$\mathbf{H} = f(\mathbf{X}) = \sigma(\mathbf{W}_e \mathbf{X} + \mathbf{b}_e),$$

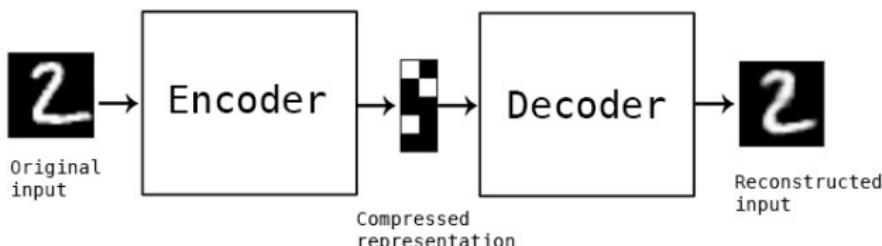
$$r(\mathbf{X}) = g(f(\mathbf{X})) = \sigma(\mathbf{W}_d \mathbf{H} + \mathbf{b}_d) \text{ — реконструкция модели.}$$

- ▶ При обучении минимизируется ошибка реконструкции:

$$\mathbb{E}_{\mathbf{X}} [\| r(\mathbf{X}) - \mathbf{X} \|_2^2],$$

- ▶ или кросс-энтропия:

$$\mathbb{E}_{\mathbf{X}} [-\mathbf{X} \log r(\mathbf{X}) - (\mathbf{I} - \mathbf{X}) \log(\mathbf{I} - r(\mathbf{X}))].$$

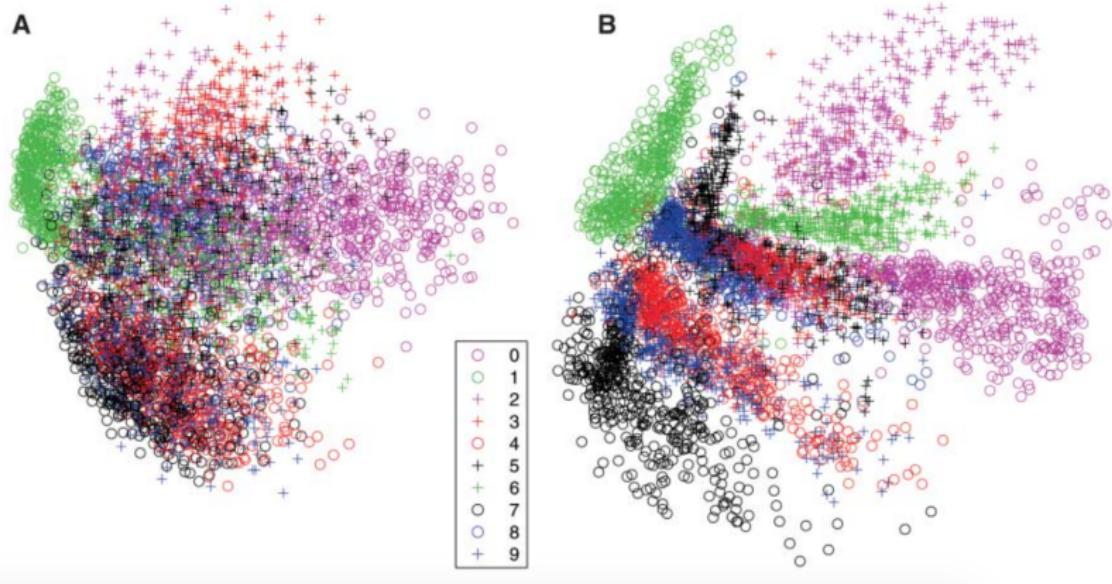


Autoencoder vs PCA



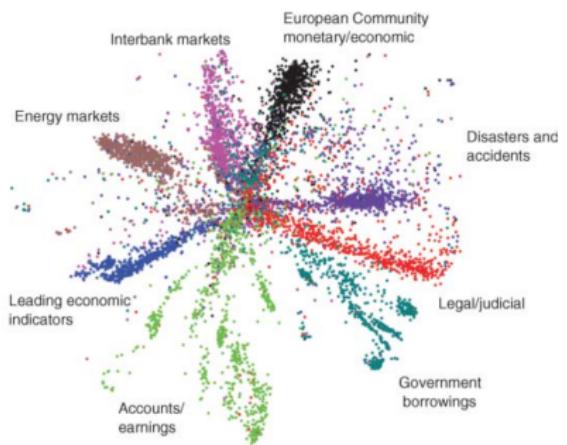
G. E. Hinton and R. R. Salakhutdinov, Reducing the Dimensionality of Data with Neural Networks — SCIENCE.

Autoencoder vs PCA



G. E. Hinton and R. R. Salakhutdinov, Reducing the Dimensionality of Data with Neural Networks — SCIENCE.

Autoencoder vs LSA



G. E. Hinton and R. R. Salakhutdinov, Reducing the Dimensionality of Data with Neural Networks — SCIENCE.

Sparse Autoencoder

Наложение ограничения разреженности на процесс реконструкции.

Разреженность — большинство компонент скрытого представления \mathbf{H} были ≈ 0 .

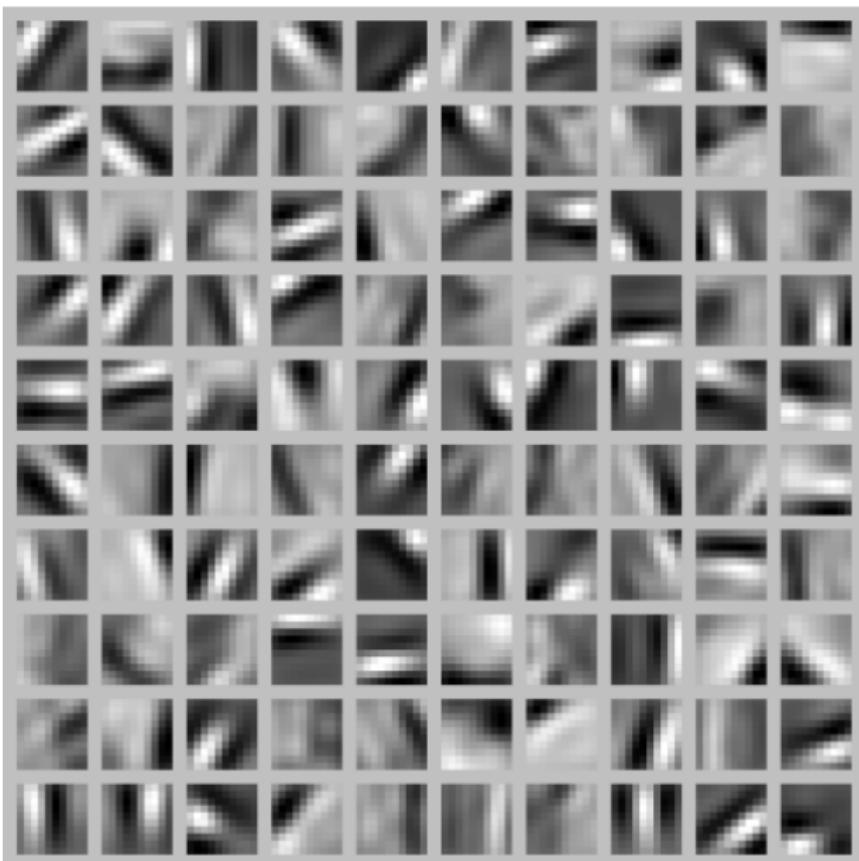
$$\hat{\rho}_j = \frac{1}{m} \sum_{i=1}^m \mathbf{h}_j(\mathbf{x}_i).$$

$\hat{\rho}_j = \rho$, где ρ — параметр разреженности(например 0.05).

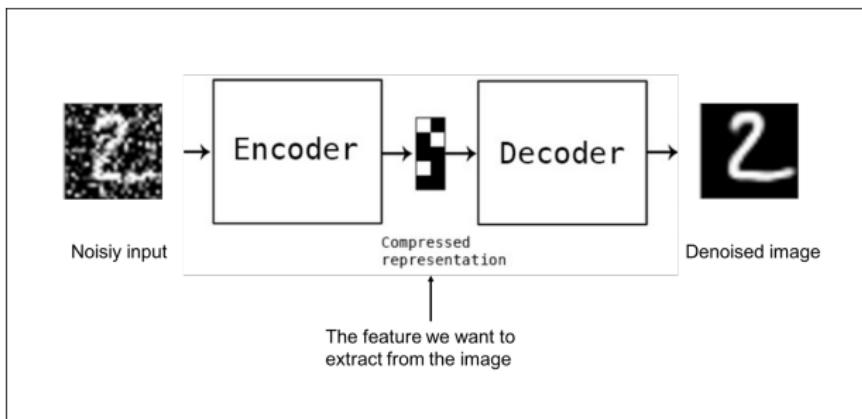
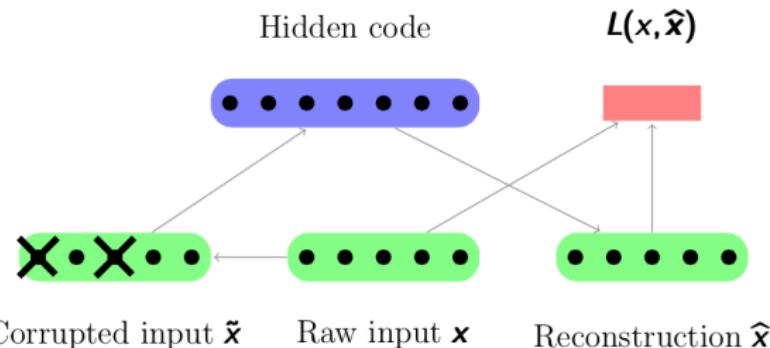
Для реализации этого ограничения к функции ошибки добавляется дополнительное штрафное слагаемое:

$$\sum_{j=1}^m KL(\rho || \hat{\rho}_j) = \sum_{j=1}^m \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1 - \rho}{1 - \hat{\rho}_j}.$$

Sparse Autoencoder



Denoising Autoencoder



Denoising Autoencoder

Denoising Autoencoder обучается, минимизируя ошибку реконструкции на зашумленном входе.

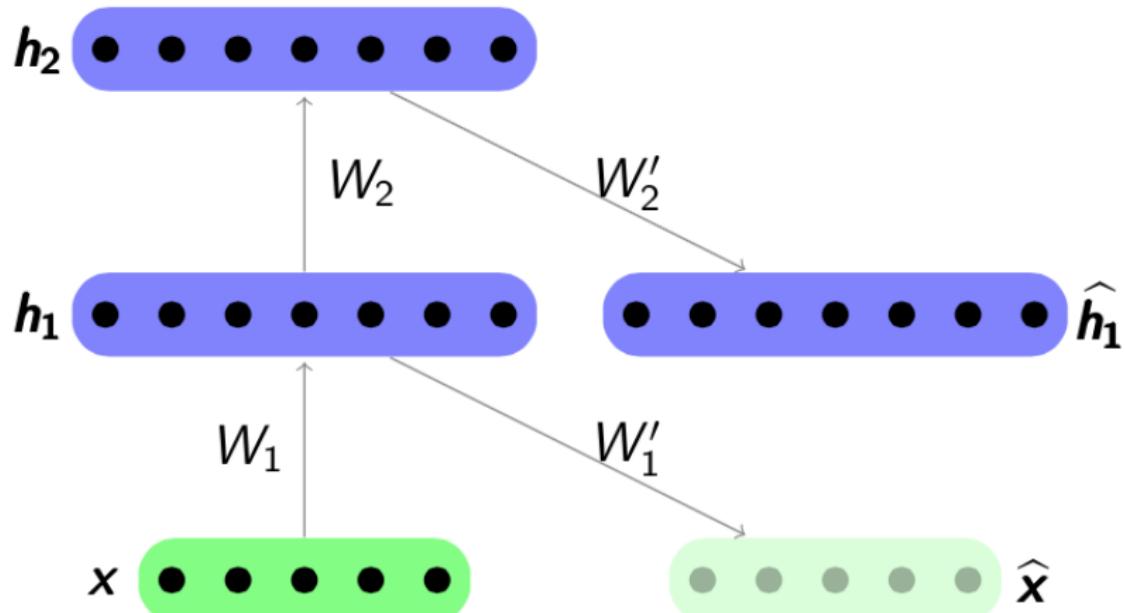
$N(\mathbf{x})$ — выберем случайно k элементов вектора \mathbf{x} и занулим их;

$$N(\mathbf{x}) = \mathbf{x} + \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 I).$$

$$\mathcal{L}_{\text{DAE}} = \mathbb{E}_{\mathbf{X}} \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 I)} \left[\| \mathbf{r}(\mathbf{X} + \boldsymbol{\epsilon}) - \mathbf{X} \|_2^2 \right].$$

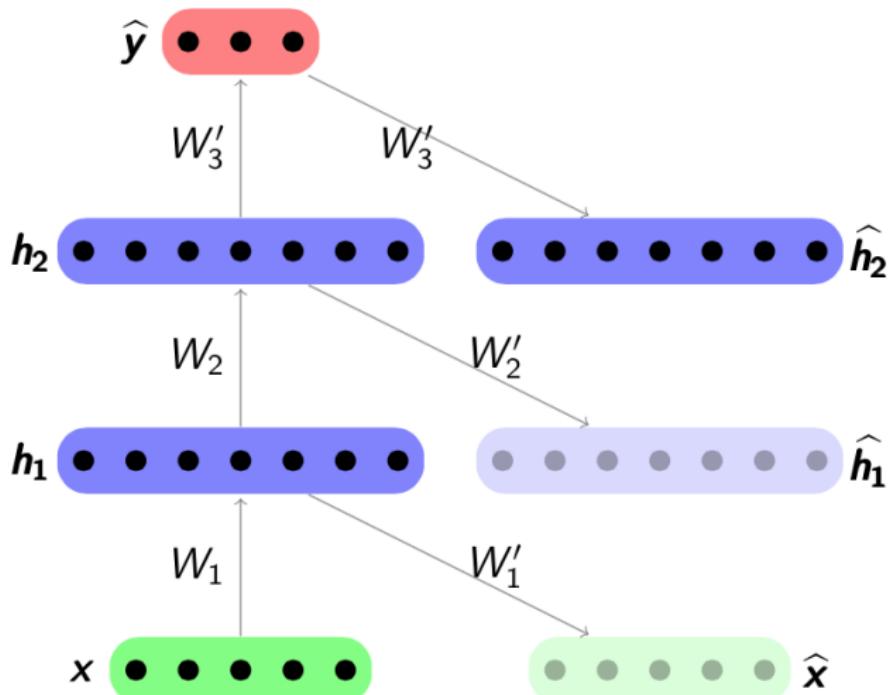
P. Vincent et al. Extracting and Composing Robust Features with Denoising Autoencoders — ICML'08.

Stacked Autoencoder



Yoshua Bengio. Learning deep architectures for AI — Foundations and Trends in Machine Learning.

Supervised fine-tuning



Yoshua Bengio. Learning deep architectures for AI — Foundations and Trends in Machine Learning.

CAE vs RCAE

- ▶ Contractive auto-encoder, CAE (Rifai et al., 2011)

$$\mathcal{L}_{\text{CAE}} = \mathbb{E}_{\mathbf{X}} \left[\| \mathbf{r}(\mathbf{X}) - \mathbf{X} \|_2^2 + \lambda \left\| \frac{\partial \mathbf{f}(\mathbf{X})}{\partial \mathbf{X}} \right\|_F^2 \right].$$

- ▶ Reconstruction contractive auto-encoder, RCAE

$$\mathcal{L}_{\text{RCAE}} = \mathbb{E}_{\mathbf{X}} \left[\| \mathbf{r}(\mathbf{X}) - \mathbf{X} \|_2^2 + \lambda \left\| \frac{\partial \mathbf{r}(\mathbf{X})}{\partial \mathbf{X}} \right\|_F^2 \right].$$



- ▶ S. Rifai et al. *Contractive auto-encoders: Explicit invariance during feature extraction* — ICML'2011.
- ▶ Guillaume Alain, Yoshua Bengio. *What Regularized Auto-Encoders Learn from the Data-Generating Distribution* — Journal of Machine Learning Research.

Теорема, Alain and Bengio.

Пусть p — дифференцируемая плотность вероятности и для $\forall \mathbf{x}_i \in \mathbb{R}^n$ $p(\mathbf{x}_i) \neq 0$. Пусть \mathcal{L}_{σ^2} — функция потерь вида:

$$\mathcal{L}_{\sigma^2} = \int_{\mathbb{R}^n} p(\mathbf{x}) \left[\| \mathbf{r}(\mathbf{x}) - \mathbf{x} \| + \sigma^2 \left\| \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right\|_F^2 \right] d\mathbf{x},$$

где \mathbf{r} дважды дифференцируема, $0 \leq \sigma \in \mathbb{R}$. Пусть $\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x})$ — оптимальная реконструкция, минимизирующая \mathcal{L}_{σ^2} . Тогда

$$\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x}) = \mathbf{x} + \sigma^2 \frac{\partial \log p(\mathbf{x})}{\partial \mathbf{x}} + o(\sigma^2), \sigma^2 \rightarrow 0.$$

Связь с энергией

$$\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x}) = \mathbf{x} + \sigma^2 \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}) + o(\sigma^2),$$

$$\frac{\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x}) - \mathbf{x}}{\sigma^2} = \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}) + o(1),$$

$$\frac{\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x}) - \mathbf{x}}{\sigma^2} \approx \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}),$$

$$\frac{\hat{\mathbf{r}}_{\sigma^2}(\mathbf{x}) - \mathbf{x}}{\sigma^2} \approx -\frac{\partial}{\partial \mathbf{x}} E(\mathbf{x}), p(\mathbf{x}) = \frac{1}{Z} \exp(-E(\mathbf{x})).$$

Векторное поле

Отображение, которое каждой точке рассматриваемого пространства ставит в соответствие вектор с началом в этой точке.

Потенциальное векторное поле

Векторное поле, которое можно представить как градиент некоторой скалярной функции.

$$\mathbf{r}(\mathbf{x}) - \mathbf{x} = \bigtriangledown E(\mathbf{x}).$$

Векторное поле

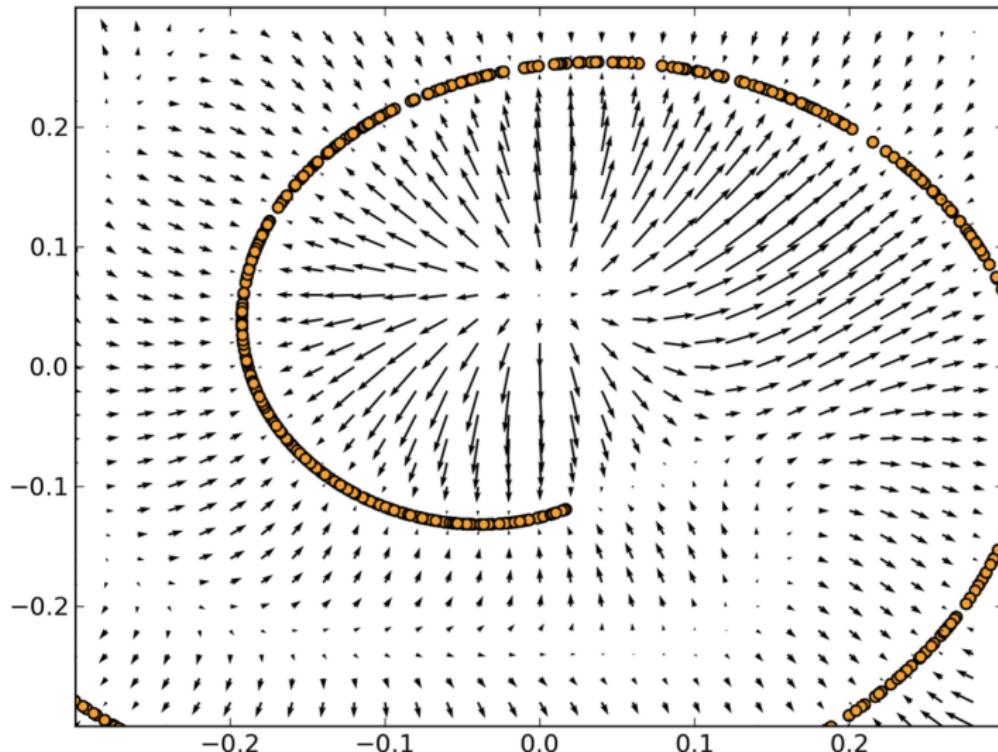
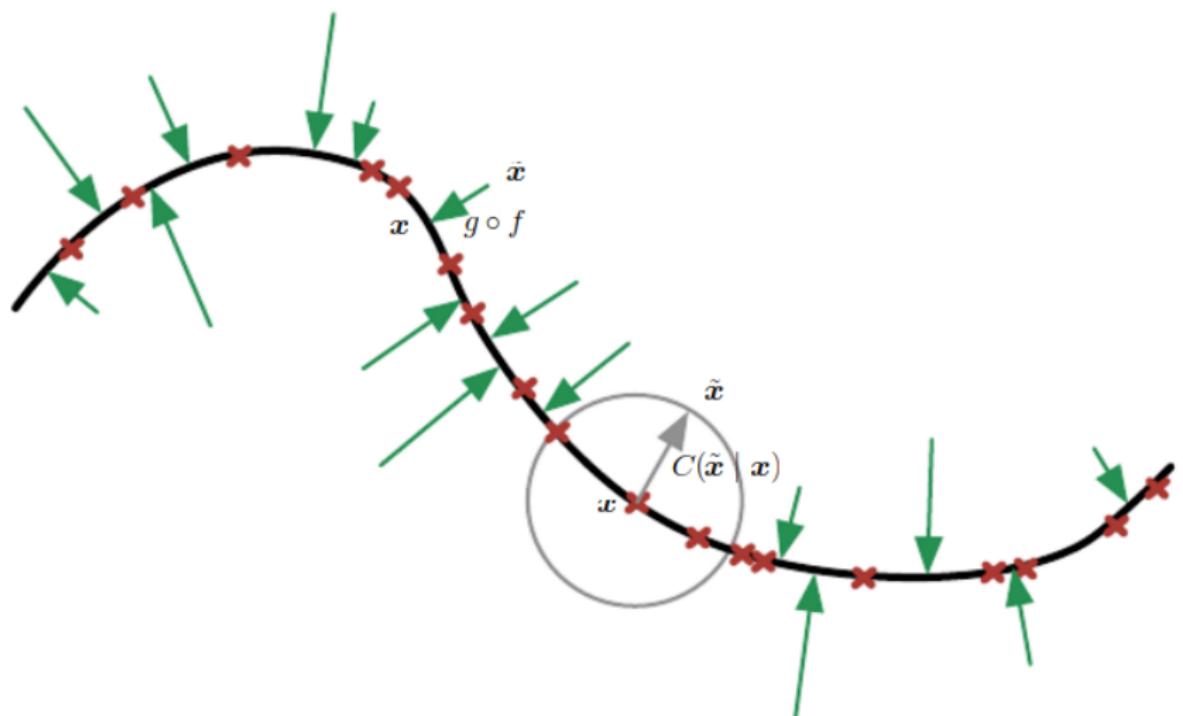
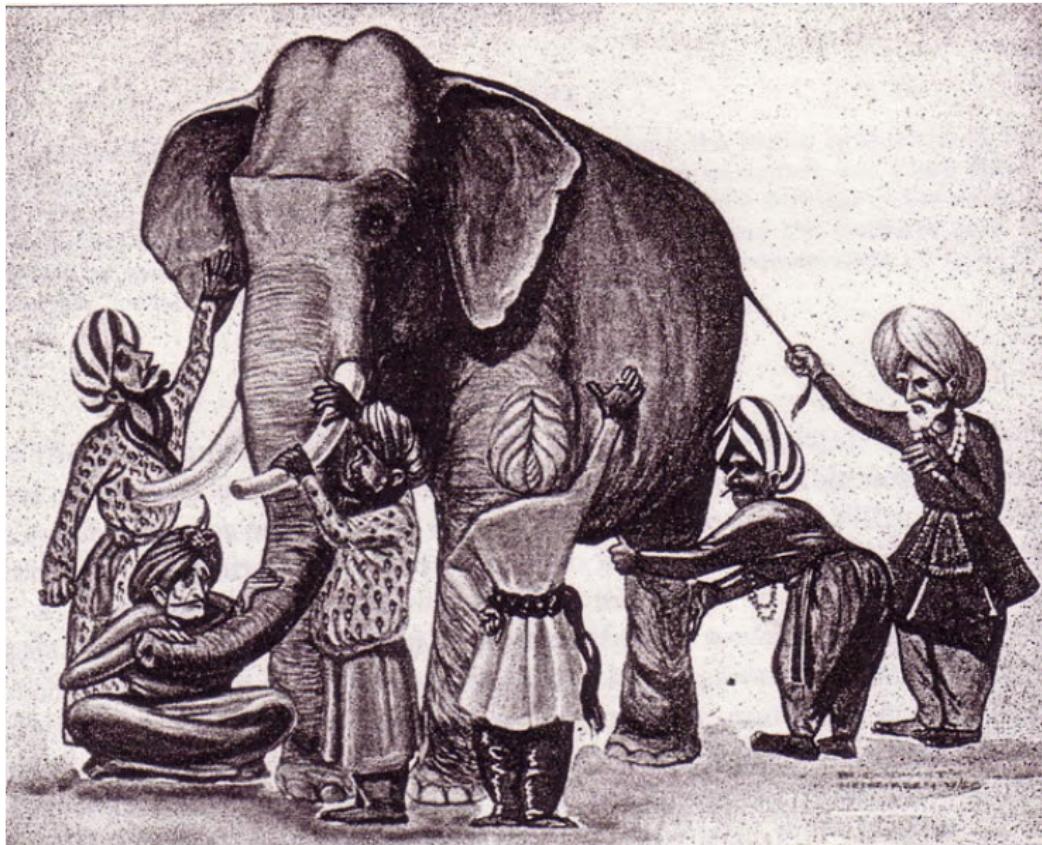


Рис.: Векторное поле $r(x) - x$.

Векторное поле Denoising Autoencoder



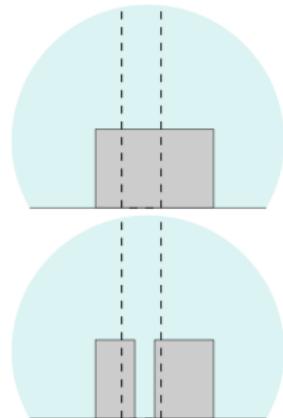


Байесовский подход

Формула Байеса:

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}$$

$$P(w|X) = \frac{P(X|w)P(w)}{\int P(X|w)p(w)dw}$$



- ▶ David J.C. MacKay, *Information Theory, Inference, and Learning Algorithms*.
- ▶ Д. Ветров, Нейробайесовский подход к задачам машинного обучения.

Вариационный автокодировщик

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

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

$p(\mathbf{h}|\mathbf{x}, \mathbf{w})$ — неизвестно.

Будем максимизировать вариационную оценку правдоподобия выборки:

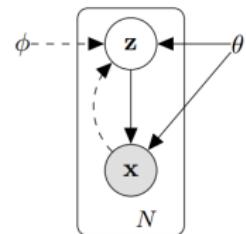
$$\log p(\mathbf{x}|\mathbf{w}) \geq E_{q_\phi(\mathbf{h}|\mathbf{x})} \log p(\mathbf{x}|\mathbf{h}, \mathbf{w}) - D_{KL}(q_\phi(\mathbf{h}|\mathbf{x}) || p(\mathbf{h})) \rightarrow \max.$$

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

$$q_\phi(\mathbf{h}|\mathbf{x}) \sim \mathcal{N}(\mu_\phi(\mathbf{x}), \sigma_\phi^2(\mathbf{x})),$$

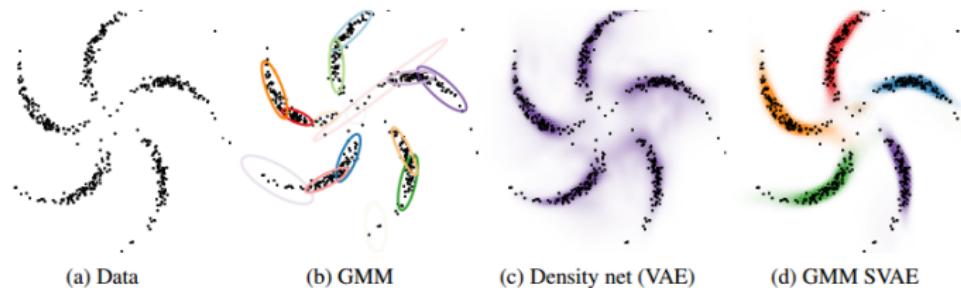
$$p(\mathbf{x}|\mathbf{h}, \mathbf{w}) \sim \mathcal{N}(\mu_w(\mathbf{h}), \sigma_w^2(\mathbf{h})),$$

где функции μ, σ — выходы нейросети.



Diederik P Kingma, Max Welling. Auto-Encoding Variational Bayes — ICLR'14.

VAE как вероятностная модель



Matthew James Johnson et al. Composing graphical models with neural networks for structured representations and fast inference — arXiv:1603.06277.

ПРИМЕРЫ ПРИКЛАДНЫХ ЗАДАЧ

- ▶ Sentiment Analysis
- ▶ Paraphrase Detection
- ▶ Learning to Rank
- ▶ Machine Translation
- ▶ Cross-lingual Plagiarism Detection
- ▶ ...

Recursive Autoencoder

Дано:

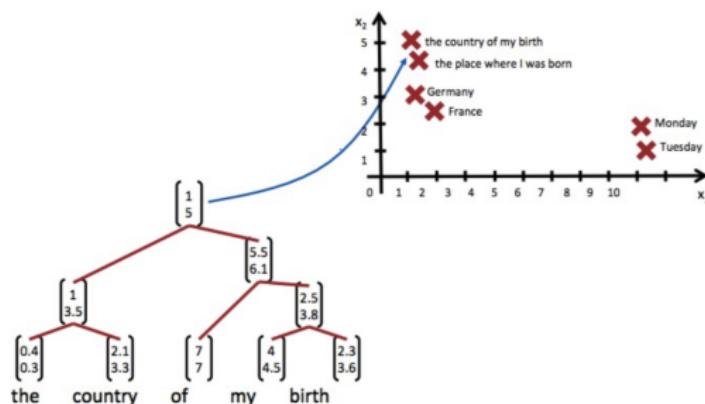
$S = \{s_i\}$ — множество предложений;

$s_i = (\mathbf{x}_1, \dots, \mathbf{x}_n)$, где $\mathbf{x}_k \in \mathbf{X}$ — вектор слова.

Требуется найти отображение:

$$\phi : s_i \rightarrow \mathbf{s}_i \in \mathbb{R}^n,$$

такое, что



Richard Socher et al. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection — NIPS'11.

Unfolding Recursive Autoencoder

Encoder:

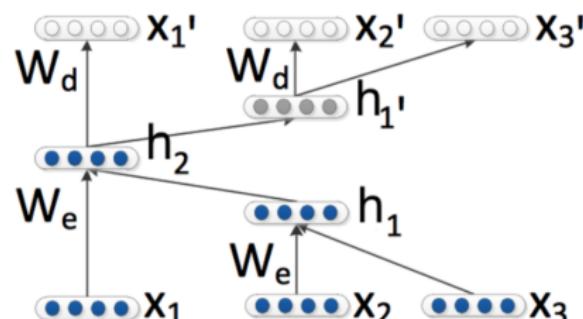
$$\mathbf{h}_1 = \sigma(\mathbf{W}_e[\mathbf{x}_2, \mathbf{x}_3] + \mathbf{b}_e),$$

$$\mathbf{h}_2 = \sigma(\mathbf{W}_e[\mathbf{x}_1, \mathbf{h}_1] + \mathbf{b}_e).$$

Decoder:

$$[\mathbf{x}'_1, \mathbf{h}'_1] = \sigma(\mathbf{W}_d \mathbf{h}_2 + \mathbf{b}_d),$$

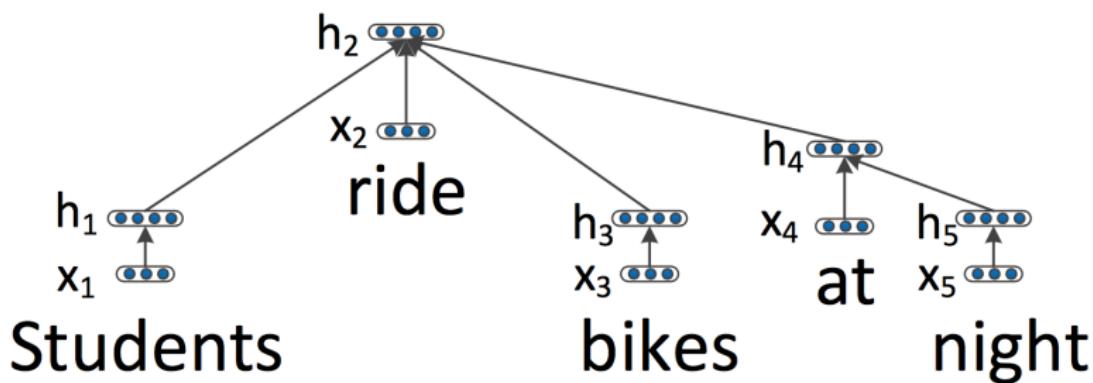
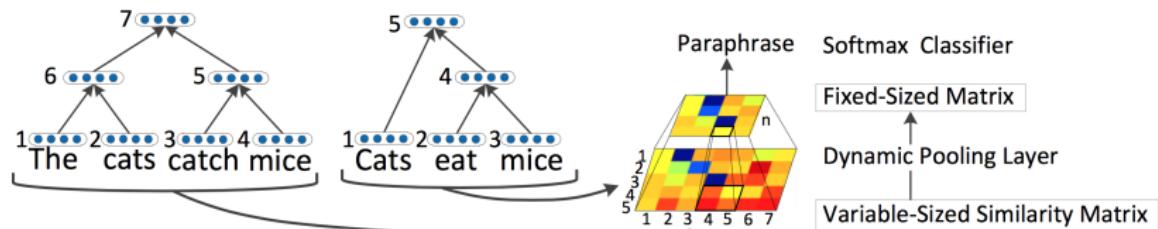
$$[\mathbf{x}'_2, \mathbf{x}'_3] = \sigma(\mathbf{W}_d \mathbf{h}_1 + \mathbf{b}_d).$$



Ошибка реконструкции:

$$\mathcal{L}_{\text{RecNN}} = \| [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3] - [\mathbf{x}'_1, \mathbf{x}'_2, \mathbf{x}'_3] \|_2^2.$$

Unfolding Recursive Autoencoder



Skip-Thought Vectors

Для тройки предложений (s_{i-1}, s_i, s_{i+1}) :

- ▶ Encoder для s_i
 - ▶ $r^t = \sigma(W_r x^t + U_r h^{t-1})$
 - ▶ $z^t = \sigma(W_z x^t + U_z h^{t-1})$
 - ▶ $\tilde{h}^t = \tanh(Wx^t + U(r^t \odot h^{t-1}))$
 - ▶ $h^t = (1 - z^t) \odot h^{t-1} + z^t \odot \tilde{h}^t$
- ▶ Decoder для s_{i-1} и s_{i+1} с учетом скрытого состояния encoder
 - ▶ $r^t = \sigma(W_r^d x^{t-1} + U_r^d h^{t-1} + C_r h_i)$
 - ▶ $z^t = \sigma(W_z^d x^{t-1} + U_z^d h^{t-1} + C_z h_i)$
 - ▶ $\tilde{h}^t = \tanh(W^d x^{t-1} + U^d(r^t \odot h^{t-1}) + Ch_i)$
 - ▶ $h_{i+1}^t = (1 - z^t) \odot h^{t-1} + z^t \odot \tilde{h}^t$
- ▶ Objective
 - ▶ $\sum_t \log P(\omega_{i+1}^t | \omega_{i+1}^{<t}, h_i) + \sum_t \log P(\omega_{i-1}^t | \omega_{i-1}^{<t}, h_i)$
 - ▶ $P(\omega_{i+1}^t | \omega_{i+1}^{<t}, h_i) \propto \exp(v_{\omega_{i+1}^t} h_{i+1}^t)$

Skip-Thought Vectors

- ▶ Encoder обучается на большом текстовом корпусе.
- ▶ Для пары векторов предложений u , v формируется признаковое пространство:

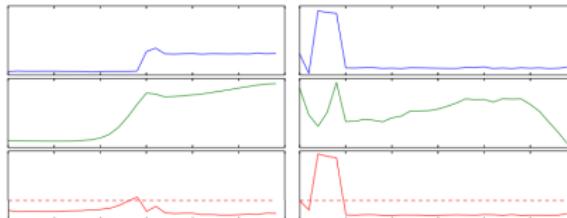
$$[u \odot v, |u - v|].$$

- ▶ На полученных признаках обучается классификатор пар.

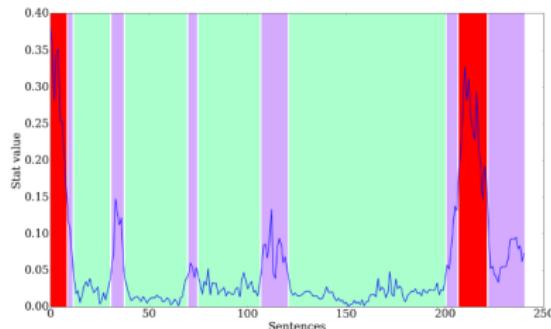
Основной недостаток — качество немного хуже конкурентов.

Основное преимущество — устойчивость к типу задачи и простая предобработка (только токенизация, в отличие от tree-LSTM, использующего парсеры, признаки с которых собираются очень долго и существуют не для всех языков).

Anomaly Detection



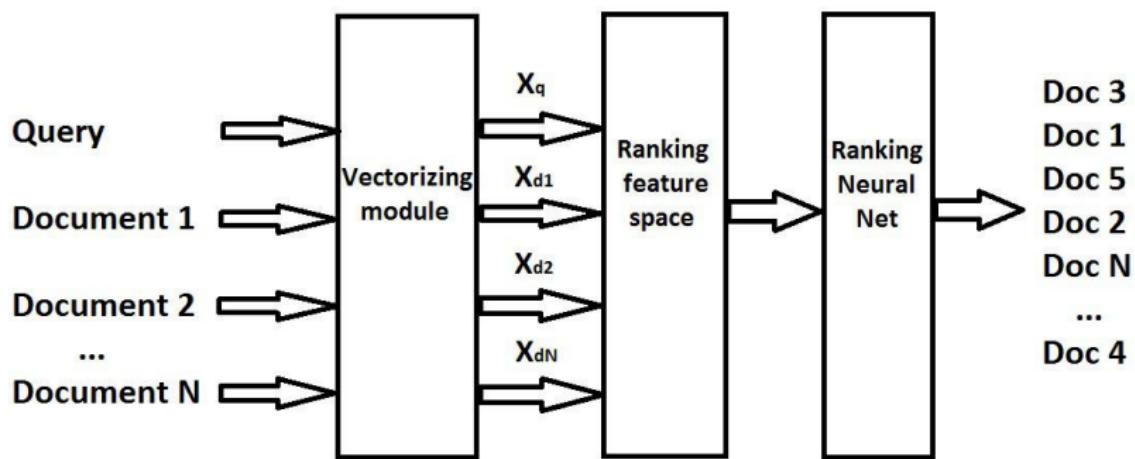
Anomaly Detection in time series



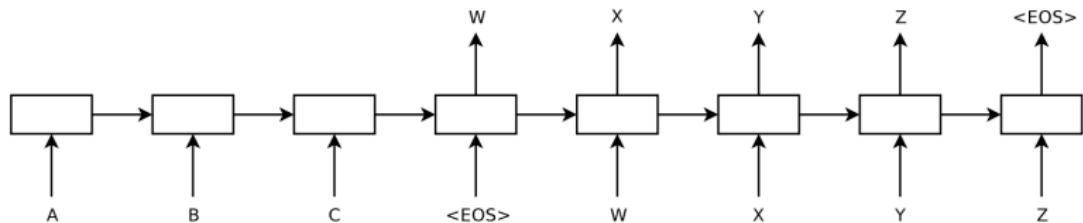
Anomaly Detection in text

- ▶ *Pankaj Malhotra et al. LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection* — arXiv:1607.00148.
- ▶ *Kamil Safin, Rita Kuznetsova, Style Breach Detection with Neural Sentence Embeddings* — CLEF'2017.

Learning to rank

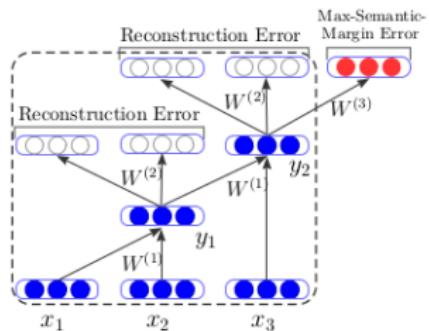
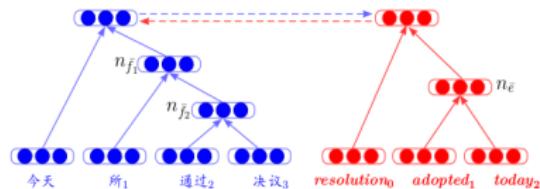


Machine translation: Seq2Seq model



Ilya Sutskever et al. Sequence to Sequence Learning with Neural Networks — NIPS'2014.

Bilingually-constrained Recursive Auto-encoder



$$E_{sem}(f, e; \theta) = E_{sem}^*(f|e, \theta) + E_{sem}^*(e|f, \theta),$$

$$E_{sem}^*(f|e, \theta) = \max\{0, E_{sem}(f|e, \theta)E_{sem}(f|e', \theta) + 1\},$$

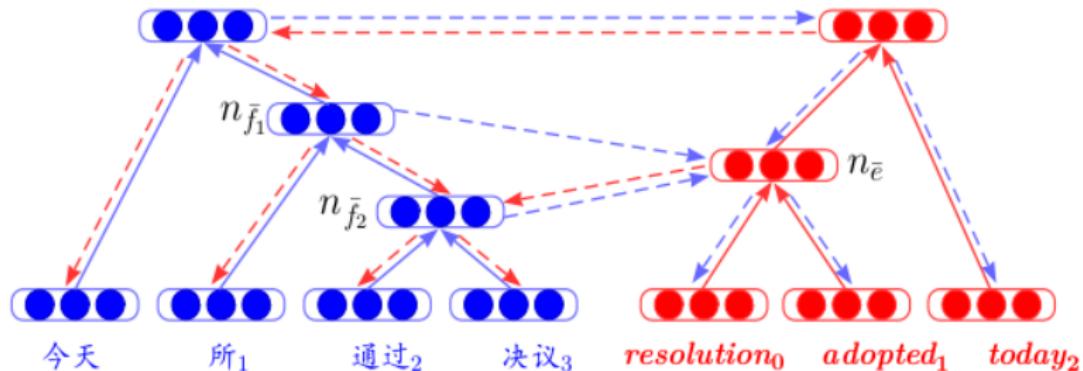
$$E_{sem}(f|e,) = \frac{1}{2} \| p_e - \tanh(W_f p_f + b_f) \|^2.$$

$$E(f, e; \theta) = \alpha(E_{rec}(f, \theta) + E_{rec}(e, \theta)) + (1 - \alpha)E_{sem}(f, e, \theta),$$

$$J_{BRAE} = \sum_{f, e \in D} E(f, e; \theta) + \frac{\lambda}{2} \| \theta \|^2.$$

J Su, D Xiong et al. Bilingual Correspondence Recursive Autoencoder for Statistical Machine Translation — EMNLP'2015.

Bilingual Correspondence Recursive Autoencoder



—→ reconstructing source sub-trees according to corresponding target nodes
—→ reconstructing target sub-trees according to corresponding source nodes



Cross-lingual Plagiarism Detection

GRU-GRU encoder:

$$\mathbf{z}_t = \sigma(\mathbf{W}_z \mathbf{x}_t + \mathbf{W}_z \mathbf{h}_{t-1} + \mathbf{b}_z),$$

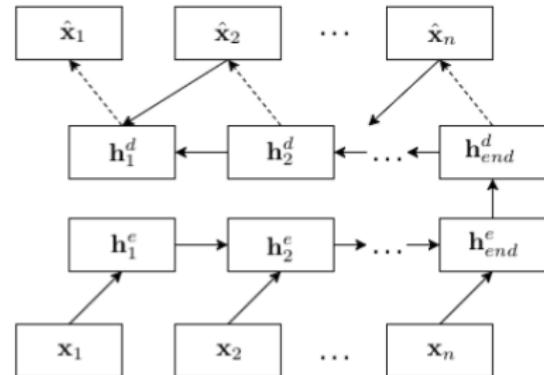
$$\mathbf{r}_t = \sigma(\mathbf{W}_r \mathbf{x}_t + \mathbf{W}_r \mathbf{h}_{t-1} + \mathbf{b}_r),$$

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_x \mathbf{x}_t + \mathbf{W}_h \mathbf{h}_{t-1} + \mathbf{b}),$$

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \circ \mathbf{h}_{t-1} + \mathbf{z}_t \circ \tilde{\mathbf{h}}_t.$$

GRU-GRU decoder:

$$\hat{\mathbf{x}}_n = \mathbf{W}_d \mathbf{h}_e^d + \mathbf{b}_d, \text{ где } \mathbf{h}_e^d = \mathbf{h}_e^d.$$



Ошибка реконструкции:

$$E_{rec} = \sum_{i=1}^n \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|_2^2.$$

Cross-lingual Plagiarism Detection

$$E_{me} = \frac{1}{|\mathcal{S}|} \left(\sum_{(\mathbf{s}_i, \mathbf{s}_j) \in \mathcal{S}} \max(0, \delta - \cos(\mathbf{s}_i, \mathbf{s}_j) + \cos(\mathbf{s}_i, \mathbf{s}_{i'})) + \right. \\ \left. + \max(0, \delta - \cos(\mathbf{s}_i, \mathbf{s}_j) + \cos(\mathbf{s}_j, \mathbf{s}_{j'})) \right),$$

δ — отступ,

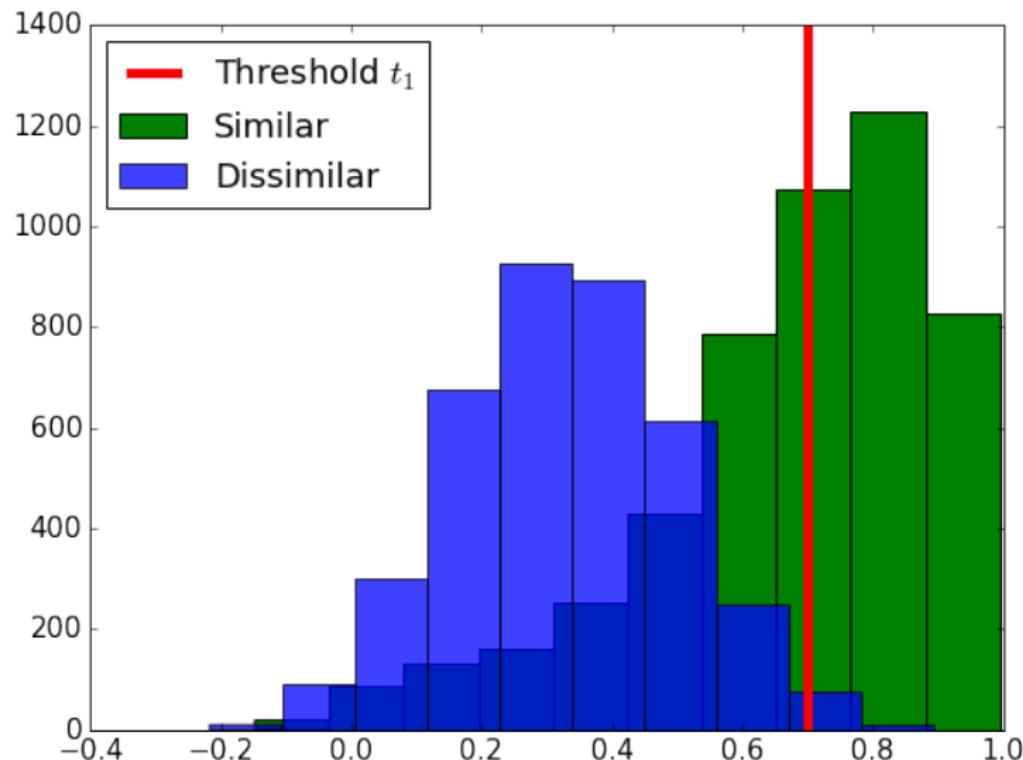
$\mathbf{s}_{i'} = \arg \max_{\mathbf{s}_{i'} \in \mathcal{S}_b \setminus (\mathbf{s}_i, \mathbf{s}_j)} \cos(\mathbf{s}_i, \mathbf{s}_{i'})$, $\mathcal{S}_b \in \mathcal{S}$ — текущий батч.

Итоговая функция ошибки:

$$\alpha E_{rec} + (1 - \alpha) E_{me} \rightarrow \min .$$

John Wieting et al. Towards universal paraphrastic sentence embeddings — arXiv:1511.08198.

Cross-lingual Plagiarism Detection



Разделимость пар предложений.

Пример

История машинного перевода.
Обнаружилось 78% заимствований с английской википедии.

Д.А. Гондоров, студ.; рук. И.В. Волкова, к. пс. н., доцент
(Филиал МЭН в г. Смоленске)

ИСТОРИЯ МАШИННОГО ПЕРЕВОДА

Машинный перевод - это под область компьютерной лингвистики, которая исследует использование программного обеспечения для перевода текста или речи с одного естественного языка на другой.

В середине 1930-х годов "машины-переводчики" были запатентованы Жоржем Арстрони, для автоматического двуязычного словаря, используя перфокарты. Петр Троянский представил более подробное предложение, включающее двуязычный словарь и способ борьбы с грамматическими различиями между языками, на основе грамматического строя эсперанто. Эта система была разделена на три этапа: на первом этапе носитель языка организовывал слова в их логических формах и проявлениях синтаксических функций; второй этап требует машины "перевести" эти формы в целевом языке; и в третьем требуется носитель языка, чтобы нормализовать этот вывод. Предложение Троянского оставалось неизвестным до конца 1950-х годов, но к тому времени компьютеры были хорошо известны и реализованы.

Первая работа по компьютерному машинному переводу были представлены в 1949 году Юренином Вивером, научным сотрудником фонда Рокфеллера, и были названа "Меморандумы перевода". Эта работа была основана на информационной теории, которая позволяла успешно взламывать



The screenshot shows the English Wikipedia article titled "History of machine translation". The page includes a navigation bar with links like "Article", "Talk", "Edit", "View history", and "Search Wikipedia". The main content discusses the origins of machine translation in the 1950s, mentioning the Georgetown experiment and its failure due to lack of funding. It highlights the development of neural machine translation in the 1980s and the rise of deep learning-based systems in the 2000s, noting the shift from rule-based to data-driven approaches. The page also covers the impact of Google Translate and other commercial systems.

Пример

История машинного перевода.
Обнаружилось 78% заимствований с английской википедии.

... французско-немецкий проект Quaero исследует возможности использования машинного перевода для многоязычного интернета. Цель проекта - перевод не только web-страниц, но и перевод видео- и аудио-файлов в интернете. В наши дни только несколько компаний используют статистический машинный перевод в коммерческих целях (продают переводы и сервисы), Гугл (использует собственные статистические системы машинного перевода, для некоторых языковых пар в Google language tools), Microsoft (использует собственные статистические системы машинного перевода для переводы базы знаний). Вновь возобновился интерес к гибридизации, к комбинированию синтаксических и морфологических (то есть, лингвистических) знаний в статистических системах ...

... French-German project Quaero investigates the possibility of making use of machine translations for a multi-lingual internet. The project seeks to translate not only webpages, but also videos and audio files on the internet. Today, only a few companies use statistical machine translation commercially, e.g. Omnisien Technologies (formerly Asia Online), SDL / Language Weaver (sells translation products and services), Google (uses its proprietary statistical MT system for some language combinations in Google's language tools), Microsoft (uses its proprietary statistical MT system to translate knowledge base articles), and Ta with you (offers a domain-adapted machine translation solution based on statistical MT with some linguistic knowledge). There has been a renewed interest in hybridisation, with researchers combining syntactic and morphological (i.e., linguistic) knowledge into statistical systems, as well as combining statistics ...

Основные библиотеки

- ▶ TensorFlow
- ▶ Theano
- ▶ Keras
- ▶ Lasagne
- ▶ ...