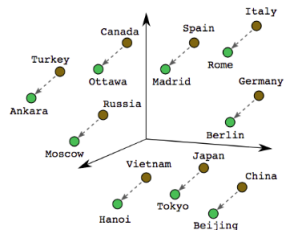
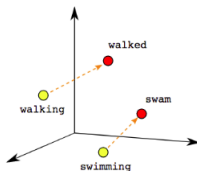
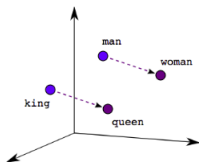


## Векторные представления объектов

Виктор Китов

[victorkitov.github.io](https://github.com/victorkitov)



# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram
- 5 Методы на основе матрицы встречаемости
- 6 Представления параграфов
- 7 Сиамская сеть

## Стандартное представление слов

- Обозначим  $V$  = размер словаря.
- Стандартные представления слов  $x \in \mathbb{R}^V$ :
  - $x_w = \mathbb{I}[w \text{ встретился в документе}]$
  - $x_w = TF_w = \#[w \text{ встретился в документе}]$
  - $x_w = TF_w IDF_w, IDF_w = \frac{N}{N_w}$ 
    - $N$  - # документов
    - $N_w$  - # документов, содержащих  $w$  хотя бы раз.
- $V$  велико, поэтому нужно компактное представление (word embedding)  $x \in \mathbb{R}^K, K \ll V$ :
  - меньше входов  $\Rightarrow$  меньше параметров  $\Rightarrow$  ниже переобучение
  - возможность учитывать семантическое сходство/различие
    - например, синонимы "автомобиль" и "машина"

# Интерпретируемые векторные представления слов

- Можно из слов извлекать интерпретируемые признаки:
  - $x^1$ : часть речи
  - $x^2$ : род (м/ж/ср - для существительных)
  - $x^3$ : время (пр/наст/буд - для глаголов)
  - $x^4$ :  $\mathbb{I}$  [начинается с заглавной буквы]
  - $x^5$ : # букв
  - $x^6$ : категория: машинное обучение, физика, биология, ...
  - $x^7$ : подкатегория: обучение с учителем, без учителя, частичное обучение, ...
  - ...
- Необходимо придумывать признаки под задачу, производить разметку.
- Легче работать с неинтерпретируемыми признаками, но которые извлекаются автоматически.

## Неинтерпретируемые представления слов

- Хотим, чтобы семантически близким словам соответствовали близкие представления.
- Дистрибутивная гипотеза (distributional hypothesis): слова близки по смыслу  $\Leftrightarrow$  они часто встречаются совместно
- "точность бустинга", "бустинг дал точность", "ниже точность, по сравнению с бустингом"
  - "точность" и "бустинг" связаны!
- Типичная размерность векторного представления  $\in [300, 500]$ .

## Представления фраз

Можно обрабатывать фразы как отдельные "слова".

- Коллокации (неслучайно часто встречающиеся слова):

$$(w_i, w_j)\text{-коллокация} \iff \frac{p(w_i w_j) - \delta}{p(w_i)p(w_j)}$$

## Представления фраз

Можно обрабатывать фразы как отдельные "слова".

- Коллокации (неслучайно часто встречающиеся слова):

$$(w_i, w_j)\text{-коллокация} \iff \frac{p(w_i w_j) - \delta}{p(w_i)p(w_j)}$$

$> threshold$ .  $\delta$  - параметр, снижающий значимость редко встречающихся слов.

# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram
- 5 Методы на основе матрицы совстречаемости
- 6 Представления параграфов
- 7 Сиамская сеть

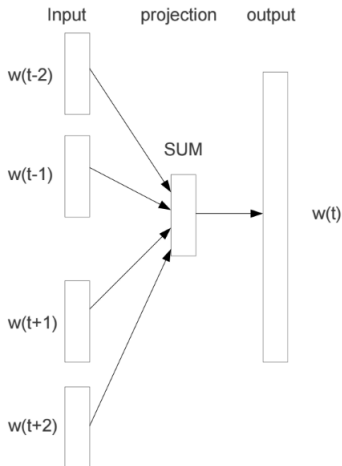


# Word2vec

- Для каждого  $w$  оценим:
  - целевое представление слова  $v_w$
  - контекстное представление слова  $\tilde{v}_w$ 
    - впоследствии можно не использовать, усреднить или конкатенировать с целевым представлением

## CBOW: идея

Continuous bag of words (CBOW): предсказываем центральное слово по контексту.



## CBOW: модель

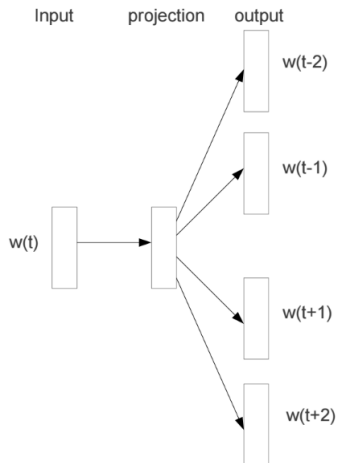
$$\frac{1}{T} \sum_{t=1}^T \ln p(w_t | w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}) \rightarrow \max_{\theta}$$

где  $\tilde{v}_{context} = \sum_{-c \leq i \leq c, i \neq 0} \tilde{v}_{w_{t+i}}$  и

$$p(w_t | w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}) = \frac{\exp(\tilde{v}_{context}^T v_{w_t})}{\sum_{w=1}^V \exp(\tilde{v}_{context}^T v_w)}$$

## Skip-gram: идея

Skip-gram: предсказываем контекст по центральному слову:



# Skip-gram: модель

$$\frac{1}{T} \sum_{t=1}^T \sum_{-c \leq i \leq c, i \neq 0} \ln p(w_{t+i} | w_t) \rightarrow \max_{\theta}$$

$$p(w_{t+i} | w_t) = \frac{\exp(\tilde{v}_{w_t}^T v_{w_{t+i}})}{\sum_{w=1}^V \exp(\tilde{v}_{w_t}^T v_w)}$$

# Комментарии

- Можем извлекать представления для др. объектов из последовательностей.
  - символы, биграммы, триграммы символов (см. *FastText*), предложения
  - нуклеотиды в ДНК последовательности
  - сервисы, заказанные клиентом компании
- Можем использовать ансамбли представлений
  - сумма, среднее, конкатенация

# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений**
- 4 Настройка skip-gram
- 5 Методы на основе матрицы встречаемости
- 6 Представления параграфов
- 7 Сиамская сеть

## Похожие слова по представлению<sup>1</sup>

- Ближайшие соседи слова пространстве эмбедингов - слова, похожие по смыслу (корпус GoogleNews, cosine-sim):
  - student -> teacher, faculty, school, university
  - car -> truck, jeep, vehicle
  - country -> nation, continent, region

---

<sup>1</sup>[http://epsilon-it.utu.fi/wv\\_demo/](http://epsilon-it.utu.fi/wv_demo/)



## Формы слов

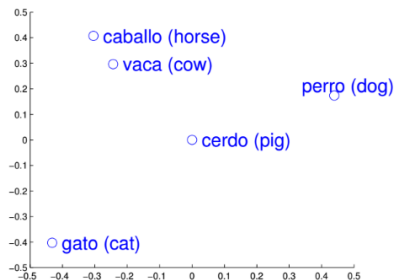
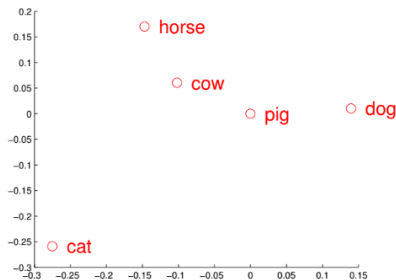
Одинаковые слова в разных формах образуют похожие структуры:



Представления могут помочь строить др. формы новых и редких слов.

# Слова на разных языках

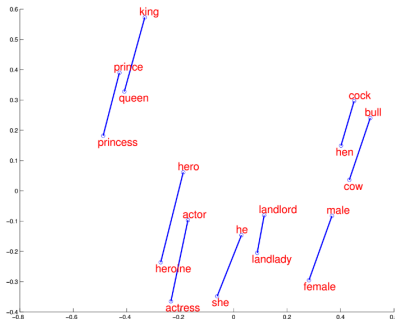
Слова на разных языках группируются похожим образом:



Представления слов могут помочь в переводе на др. язык.

# Семантическая регулярность

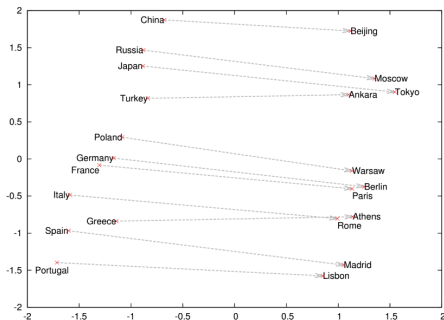
Слова, связанные семантически определенным образом группируются единообразно:



$(\text{prince} - \text{princess}) + \text{queen} \approx \text{king}$ . Может помочь в системе автоматических ответов на вопросы.

# Семантическая регулярность

Слова, связанные семантически определенным образом группируются единообразно:



(Beijing-China)+Russia $\approx$ Moscow! Может помочь в системе автоматических ответов на вопросы.

# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram**
- 5 Методы на основе матрицы совстречаемости
- 6 Представления параграфов
- 7 Сиамская сеть

## Вычислительная сложность Word2vec

$$\text{CBOW: } \frac{1}{T} \sum_{t=1}^T \ln p(w_t | w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}) \rightarrow \max_{\theta}$$

где  $\tilde{v}_{context} = \sum_{-c \leq i \leq c, i \neq 0} \tilde{v}_{w_{t+i}}$  и

$$p(w_t | w_{t-c}, \dots, w_{t-1}, w_{t+1}, \dots, w_{t+c}) = \frac{\exp(\tilde{v}_{context}^T v_{w_t})}{\sum_{w=1}^V \exp(\tilde{v}_{context}^T v_w)}$$

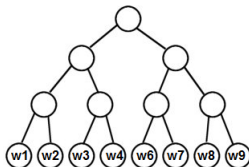
$$\text{SkipGram: } \frac{1}{T} \sum_{t=1}^T \sum_{-c \leq i \leq c, i \neq 0} \ln p(w_{t+i} | w_t) \rightarrow \max_{\theta}$$

$$p(w_{t+i} | w_t) = \frac{\exp(\tilde{v}_{w_t}^T v_{w_{t+i}})}{\sum_{w=1}^V \exp(\tilde{v}_{w_t}^T v_w)}$$

Проблема: знаменатель вычисляется за  $O(V)$ .

# Иерархический SoftMax

- Строим бинарное дерево
  - предсказываемые слова языка - листья.
- Для каждого узла  $j$ :



$$p(\text{left}|j) = \sigma \left( v_j^T \tilde{v}_{\text{context}} \right),$$
$$p(\text{right}|j) = 1 - \sigma \left( v_j^T \tilde{v}_{\text{context}} \right) = \sigma \left( -v_j^T \tilde{v}_{\text{context}} \right)$$

- $p(w|\text{context})$  = произведение вероятностей дойти до него.
- Сложность вычисления  $p(w_i|\text{context})$ :  $O(V) \rightarrow O(\log_2 V)$

## Иерархический SoftMax

- Целевое представление  $w_i$  теперь не вектор, а  $\{v_j\}_j$  для всех узлов  $j$  дерева.
- Эффективнее работает не сбалансированное дерево, а дерево Хаффмана
  - более частотным словам - более короткие пути.



## Негативное сэмплирование

- Негативное сэмплирование (negative sampling)<sup>2</sup> - аппроксимация максимизация правдоподобия.
- Для каждой реальной (позитивной) пары  $(w_t, w_{t+i})$  сэмплируем  $K$  негативных случайно  $(w_t, w_{j(1)}), \dots (w_t, w_{j(K)})$ .

$$\underbrace{\ln \left( \frac{1}{1 + e^{-\tilde{v}_{w_t}^T v_{w_{t+i}}}} \right)}_{\sigma(+\tilde{v}_{w_t}^T v_{w_{t+i}})} + \sum_{k=1}^K \underbrace{\ln \left( \frac{1}{1 + e^{+\tilde{v}_{w_t}^T v_{w_{t+i}}}} \right)}_{\sigma(-\tilde{v}_{w_t}^T v_{w_{t+i}})} \rightarrow \max_{\tilde{v}, v}$$

- $K \sim 2-5$ .  $p(w_{j(k)}) \propto p(w)^{3/4}$ .

<sup>2</sup>Distributed Representations of Words and Phrases

fastText<sup>3</sup>

- Работает как skip-gram (предсказываем слова контекста по центральному слову)
- Раньше совместимость была  $\tilde{w}_t^T w_{t+i}$
- В fastText  $\tilde{w}'_t := \tilde{w}_t + \sum_{p \in n\text{-grams}(w_t)} \text{part}_p$ ,  $w_{t+i}$  - прежняя.
- Новая совместимость

$$\tilde{w}_t^T w_{t+i} + \sum_{p \in n\text{-grams}(w_t)} \text{part}_p^T w_{t+i}$$

- Пример триграмм: "person" ->  
 "<person>", "<pe", "per", "ers", "rso", "son", "on>"
  - предлагается использовать все n-граммы,  $3 \leq n \leq 6$ .
- Работает для слов вне словаря, для слов с опечатками за счет  $\sum_{p \in n\text{-grams}(w)} \text{part}_p$ .

<sup>3</sup><https://arxiv.org/pdf/1607.04606.pdf>, код и данные: [fasttext.cc](https://fasttext.cc)

# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram
- 5 Методы на основе матрицы совстречаемости**
- 6 Представления параграфов
- 7 Сиамская сеть

## Матрица совстречаемости слов

- $X \in \mathbb{R}^{V \times V}$  - матрица со-встречаемости слов (word co-occurrence matrix)
- $X_{ij} = \#\{\text{слово } j \text{ встретилось в контексте слова } i\}$ .
- Пример для контекста  $\pm 1$  слово:

I like deep learning. I like NLP. I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

## Разложение матрицы совстречаемости

- Hyperspace Analogue to Language (HAL)<sup>4</sup>: эмбединги из низкорангового разложения
  - напр. строки  $U$  либо столбы  $V^T$  из SVD.
  - эмбединги доминируются частыми словами!

---

<sup>4</sup>Lund and Burgess, 1996.

<sup>5</sup>Bullinaria and Levy, 2007

<sup>6</sup><https://aclanthology.org/D14-1162.pdf>

## Разложение матрицы совстречаемости

- Hyperspace Analogue to Language (HAL)<sup>4</sup>: эмбединги из низкорангового разложения
  - напр. строки  $U$  либо столбы  $V^T$  из SVD.
  - эмбединги доминируются частыми словами!
- Модификация<sup>5</sup>: счётчик совстречаемости  $\rightarrow$  PPMI

$$PPMI(w_1, w_2) = \max\{0, PMI(w_1, w_2)\} = \max\{0, \ln \frac{P(w_1, w_2)}{P(w_1)P(w_2)}\}$$

---

<sup>4</sup>Lund and Burgess, 1996.

<sup>5</sup>Bullinaria and Levy, 2007

<sup>6</sup><https://aclanthology.org/D14-1162.pdf>

## Разложение матрицы совстречаемости

- Hyperspace Analogue to Language (HAL)<sup>4</sup>: эмбединги из низкорангового разложения
  - напр. строки  $U$  либо столбы  $V^T$  из SVD.
  - эмбединги доминируются частыми словами!
- Модификация<sup>5</sup>: счётчик совстречаемости  $\rightarrow$  PPMI

$$PPMI(w_1, w_2) = \max\{0, PMI(w_1, w_2)\} = \max\{0, \ln \frac{P(w_1, w_2)}{P(w_1)P(w_2)}\}$$

- GloVe<sup>6</sup>: матр. факторизация  $\log X \approx W^T \tilde{W} + B + \tilde{B}$

$$\sum_{i,j=1}^V f(X_{ij}) \left( w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij} \right)^2 \rightarrow \min_{w, \tilde{w}, b, \tilde{b}}$$

$f(X_{ij})$  некоторая  $\uparrow$  функция весов,  $f(0) = 0$ .

<sup>4</sup>Lund and Burgess, 1996.

<sup>5</sup>Bullinaria and Levy, 2007

<sup>6</sup><https://aclanthology.org/D14-1162.pdf>

# Содержание

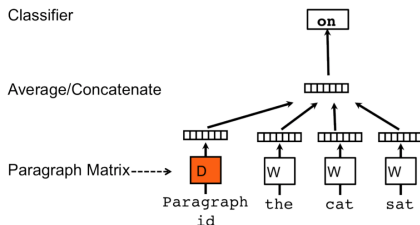
- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram
- 5 Методы на основе матрицы совстречаемости
- 6 Представления параграфов**
- 7 Сиамская сеть



## Представления параграфов - мотивация

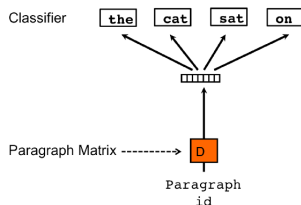
- Необходимо получить векторные представления параграфов (документов, предложений,...).
- Простой подход: усреднить слова, входящие в параграф.
  - или взвешенно усреднить, учитывая частоту встречаемости слов и их тематику.
- Точнее работает непосредственное представление самих параграфов.

# Paragraph vector: модель PV-DM



- Во время обучения делим документы на параграфы. Каждому параграфу -> векторное представление.
- Оценивается CBOW, контекст: представление слов и параграфов.
- Можно усреднять или конкатенировать контексты слов и параграфа.
- Называется *Distributed Memory Model of Paragraph Vectors (PV-DM)*.

## Paragraph vector: модель PV-DBOW



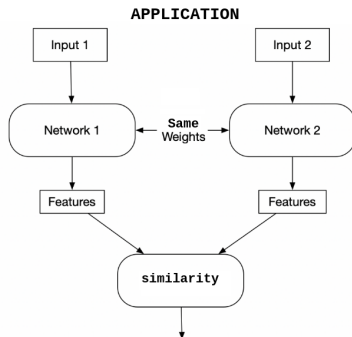
- Во время обучения делим документы на параграфы. Каждому параграфу -> векторное представление.
- Оценивается skip-gram: предсказываются случайные слова параграфа по представлению параграфа.
  - контекст: только представления параграфов
- Называется *Distributed Bag of Words version of Paragraph Vector (PV-DBOW)*

# Содержание

- 1 Векторное представление слов
- 2 Word2vec
- 3 Регулярности в пространстве представлений
- 4 Настройка skip-gram
- 5 Методы на основе матрицы совстречаемости
- 6 Представления параграфов
- 7 Сиамская сеть**

# Сиамская сеть

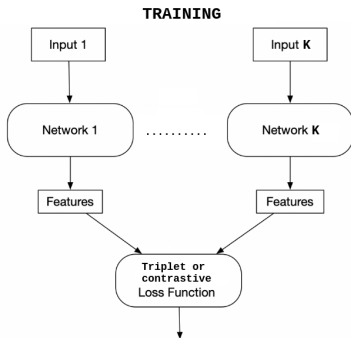
- Сиамская сеть (siamese network) генерирует 2 представления произвольных объектов.
  - объекты могут быть разных доменов.
- Обучение: похожие объекты  $\Rightarrow$  похожие представления.
  - похожесть:  $\|\cdot\|_2^2$ ,  $\langle \cdot, \cdot \rangle$ ,  $\cos$  —  $sim$



## Примеры приложений

- Классификация:
  - вход: 2 объекта (обучение) или тестовый объект (применение)
  - выход: класс на основе близости к центроиду класса или по K-NN
- Поисковая система
  - вход: документ и поисковый запрос
    - возможен и поиск по изображению
  - выход: степень релевантности документа запросу
- Обнаружение перефразирования:
  - вход: 2 предложения
  - выход: насколько они близки по смыслу
- Проверка подписи
  - вход: сканы 2х подписей
  - выход: степень их принадлежности одному человеку

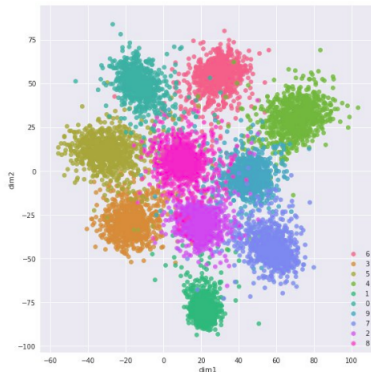
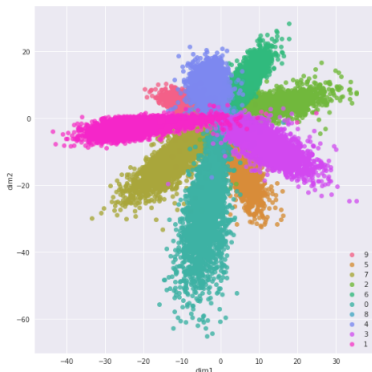
# Обучение



- Идея функции потерь:
  - представления похожих объектов должны быть близки
  - представления различных объектов должны быть далеки

# Представления объектов: классификация и сиамская сеть

Представления объектов: классификация и сиамская сеть для MNIST:





## Применение

После того, как сиамская сеть настроена, можно решать конечную задачу. Если классификация, то можно

- инициализировать классификационную сеть первыми слоями сиамской сети (особенно для instance discrimination)
- решать классификацию в пространстве эмбедингов.
  - метод ближайших центроидов
  - метод K ближайших соседей

## Попарные потери

Попарные потери (pairwise contrastive loss, spring loss)<sup>7</sup>:

- обучение на случайных парах объектов  $x_i, x_j$

$$\mathcal{L}(x_i, x_j) = \begin{cases} \rho(x_i, x_j)^2, & \text{если } x_i, x_j \text{ похожи} \\ \max\{0, \alpha - \rho(x_i, x_j)\}^2, & \text{если } x_i, x_j \text{ непохожи} \end{cases}$$

- $\alpha > 0$  - гиперпараметр (мин. расстояние для непохожих объектов, когда не будет штрафа)
- $\rho(x, x')$  - обычно Евклидово
- число уникальных пар -  $O(N^2)$ .

---

<sup>7</sup>Выгоднее позволять похожим объектам небольшую вариацию в эмбедингах. Предложите соответствующее изменение.

# Тройные потери

## Тройные потери (triplet loss):

- обучение на случайных тройках  $x, x^+, x^-$ .
- $x$  - опорный объект (anchor)
- $x^+$  - похожий на  $x$  (positive)
- $x^-$  - не похожий на  $x$  (negative)
- $\alpha > 0$  - гиперпараметр (мин. разница расстояний без штрафа)
- $\rho(x, x')$  - обычно Евклидово
$$\mathcal{L}(x, x^+, x^-) = \max \{ \rho(x, x^+)^2 - \rho(x, x^-)^2 + \alpha; 0 \}$$
- число уникальных пар -  $O(N^3)$ .

## Вероятностные потери

- Вероятностные потери (InfoNCE loss, NCE=noise contrastive estimation)
- $x$  - опорный объект (anchor)
- $x^+$  - похожий на  $x$  (positive)
- $x_1, \dots, x_M$  - набор непохожих на  $x$  объектов

$$\mathcal{L}(x, x^+, x_1^-, \dots, x_M^-) = -\ln \frac{e^{\text{sim}(x, x^+)}}{e^{\text{sim}(x, x^+)} + \sum_{m=1}^M e^{\text{sim}(x, x_m^-)}}$$

$$\text{sim}(x, x') = \frac{x^T x'}{\|x\| \cdot \|x'\|}$$

- $> O(N^3)$  уникальных сэмплов.

## Комментарии

- Сэмплировать можно равномерно
  - по объектам (максимизируем микро-усредненные метрики per object)
  - по классам (максимизируем макро-усредненные метрики per class)
- Принципы контрастного обучения можно использовать для metric learning  $\rho_{\theta}(x, x')$ .

# Сиамская сеть и классификация

- Классификация
  - выучивает "что представляет каждый класс".
  - выдает степени соответствия  $x$  каждому классу.
- Сиамская сеть
  - выучивает "что отличает классы друг от друга".
  - выдает расстояния от  $x$  до каждого класса.
  - более устойчива к дисбалансу классов и редким классам (*one shot learning*)
    - при обучении каждый класс учитывается поровну
    - модель выучивает признаки, по которым можно судить о сходстве классов на частотных классах, потом сразу подхватывает их для редких.
  - извлекает больше информации из выборки
    - обучение не на объектах, а на парах и тройках объектов.
  - хороша в ансамбле с классификатором ( $\uparrow$ разнообразие)

## Заключение

- **Представления слов** отображают слова в компактные векторные представления.
  - может применяться
    - к биграммам, триграммам, коллокациям.
    - к символам - удобно для новых слов
    - к любым объектам из посл-тей (нуклеотиды в ДНК и др.)
- **Представления параграфов** отображают параграфы в векторные представления.
  - работают лучше, чем усреднение слов параграфа
- Представления можно находить для целевой или связанной задачи (language modeling, transfer learning)
- **Сиамская сеть** оценивает похожесть пар объектов.
  - применения: классификация (особенно one shot learning), детекция перефразирования, нахождение похожих изображений, ...