ТЕХНОСФЕРА

Лекция
Word Embeddings

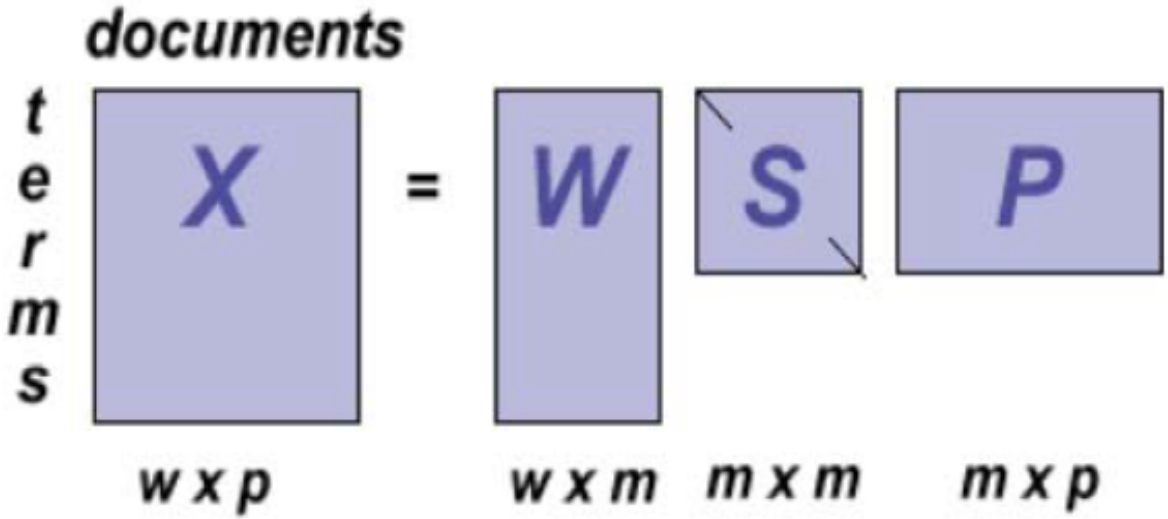
Владимир Гулин

октябрь 2020 г.

Недостатки рассмотренных моделей

Какие недостатки есть в тех моделях, которые рассмотрели?

Latent semantic analysis (1988)



$$\hat{D} = A^T D, \quad \hat{Q} = A^T Q,$$

$$A = W_k S^{-1}$$

$$sim(Q, D) = \frac{\hat{Q}\hat{D}}{\|Q\|\|D\|}$$

Latent semantic analysis (1988)

Фактически латентно-семантический анализ - это применение SVD разложения к матрице “термин-документ”

- ✔ Оценка близости документов
- ✔ Оценка близости терминов
- ✔ Кластеризация документов
- ✔ Взвешивание пары запрос-документ
- ✖ Низкая скорость для больших коллекций

Тематическое моделирование

Что такое тема?

- тема - семантический кластер текстов
- тема - набор терминов предметной области
- тема - условное распределение на множестве слов

$$p(w|t) - \text{вероятность слова } w \text{ в теме } t$$

- тема - тематический профиль документа

$$p(t|d) - \text{вероятность темы } t \text{ в документе } d$$

Цель тематической модели:

Найти латентные темы документов коллекции по наблюдаемым распределениям слов $p(w|d)$ в документах.

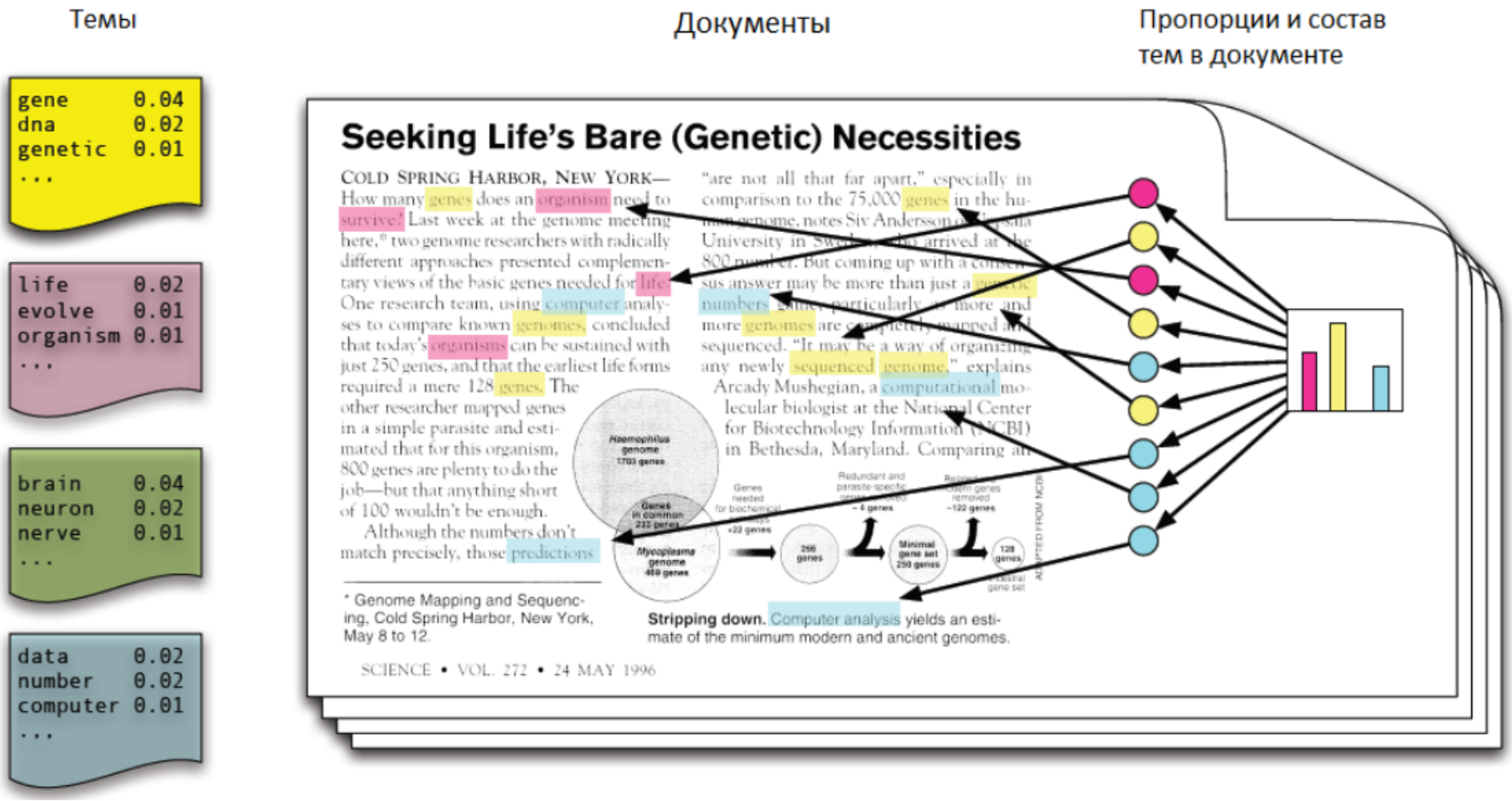
Тематическое моделирование

Основные положения:

- Модель мешка слов для документов (порядок не важен)
- Модель мешка документов для коллекции (порядок не важен)
- Коллекция - это i.i.d. выборка $(d_i, w_i, t_i)_{i=1}^n \sim p(d, w, t)$
- d_i, w_i - наблюдаемые переменные, t_i - скрытые
- Гипотеза условной независимости: $p(w|d, t) = p(w|t)$
- Считаем, что тексты предобработаны (стемминг, лемматизация, удаление стоп-слов и т.д.)

Тематическое моделирование

$$p(w|d) = \sum_{t \in T} p(w|t) \cdot p(t|d)$$



Тематическое моделирование

Дано:

- W - словарь слов
- D - коллекция документов
- $d = \{w_1 \dots w_{n_d}\}$ - документ
- n_{dw} - число раз, когда слово w встретилось в документе d
- n_d - длина документа d

Найти:

Параметры модели $\frac{n_{dw}}{n_d} \approx p(w|d) = \sum_{t \in T} \psi_{wt} \theta_{td}$

$\psi_{wt} = p(w|t)$ - вероятности слов w в каждой теме t

$\theta_{td} = p(t|d)$ - вероятности тем w в каждой документе d

Тематическое моделирование

- $X = (d_i, w_i)_{i=1}^n$ - исходные данные
- $T = (t_i)_{i=1}^n$ - скрытые переменные, темы
- $\Omega = (\Psi, \Theta)$ - параметры

Нужно по X найти Ω

Максимизируем неполное правдоподобие

$$ln\, p(X|\Omega) = ln \sum_T p(X, T|\Omega) \to \max_\Omega$$

EM алгоритм:

$$\text{E-step:} \quad q(T) = p(T|X, \Omega)$$

$$\text{M-step:} \quad \sum_T q(T) ln\, p(X, T|\Omega) \to \max_\Omega$$

Тематическое моделирование

$p(\Omega)$ - априорное распределение параметров модели

Принцип максимума правдоподобия

$$p(X, \Omega) = p(X|\Omega)p(\Omega) \to \max_{\Omega}$$

$$ln\, p(X, \Omega) = ln\, p(X|\Omega) + \ln\, p(\Omega) \to \max_{\Omega}$$

Обозначим $R(\Omega) = \ln\, p(\Omega)$

PLSA [Hofmann, 1999]: $R(\Omega) = 0$

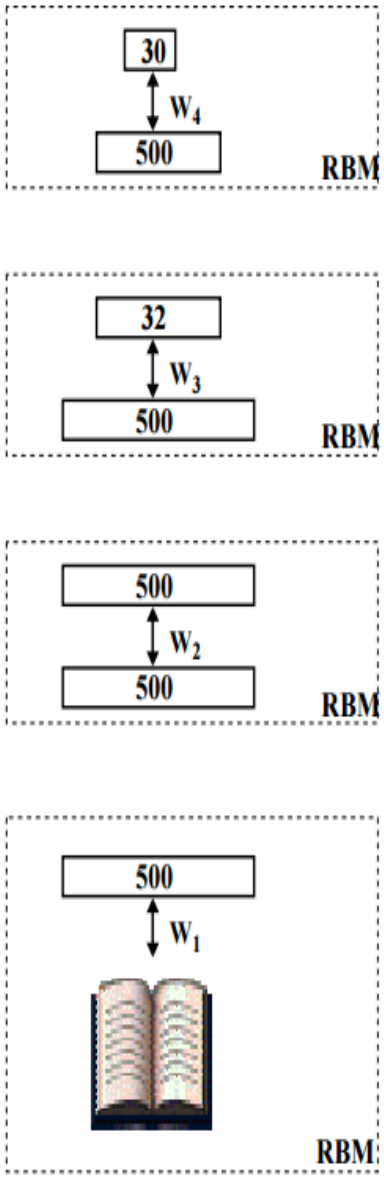
LDA [Blei, 2003]: $R(\Omega) = ln \prod_{t \in T} Dir(\psi_t|\beta) \prod_{d \in D} Dir(\theta_d|\alpha)$

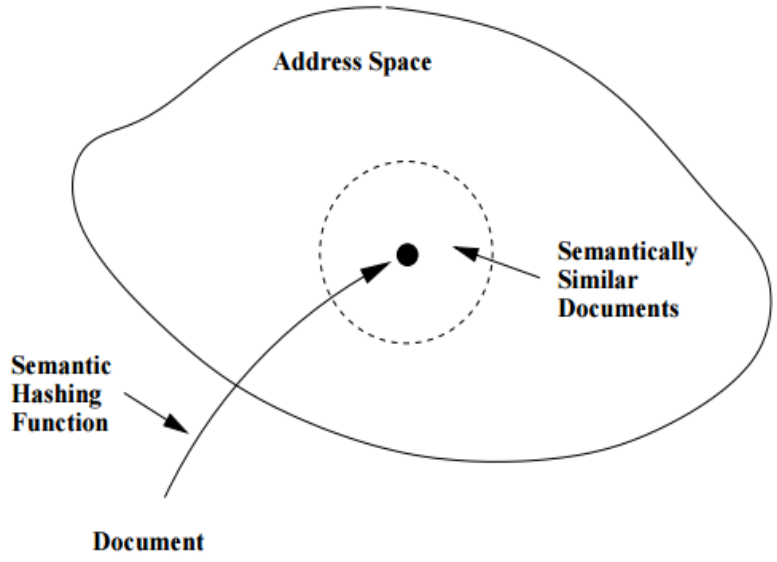
EM алгоритм:

$$\text{E-step:} \quad q(T) = p(T|X, \Omega)$$

$$\text{M-step:} \quad \sum_{T} q(T) ln\, p(X, T|\Omega) + R(\Omega) \to \max_{\Omega}$$

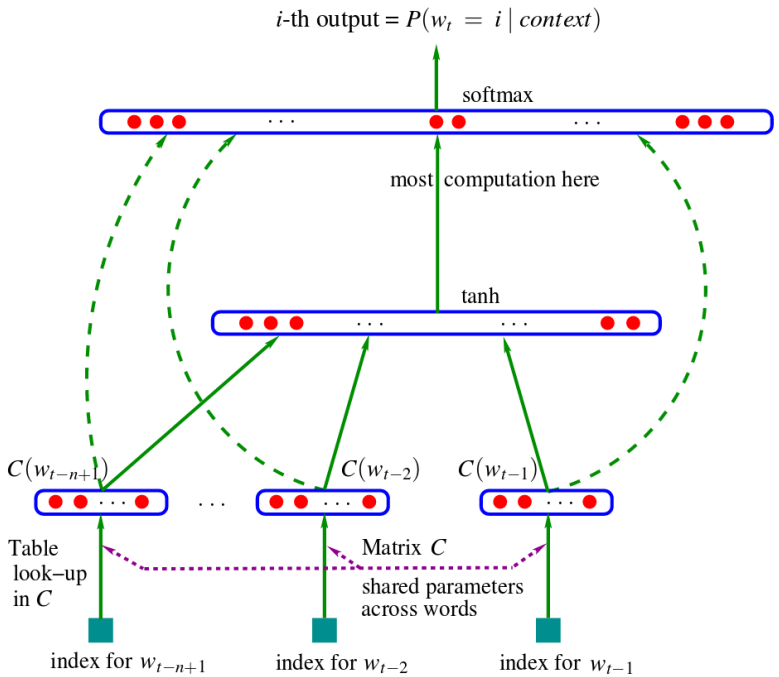
Semantic hashing (Hinton, Salakhutdinov, 2009)





A Neural Probabilistic Language Model (Y. Bengio 2003)

Пытаемся с помощью нейросетей оценить вероятность следующего слова по набору из предыдущих слов (сеть может быть как последовательной так и рекурентной)



✖ Долго и сложно обучать

Дистрибутивная гипотеза

Гипотеза:

Лингвистические единицы, встречающиеся в схожих контекстах, имеют близкие значения.

Вывод:

Значит, векторы слов, можно построить с помощью контекстов этих слов.

Представление слов контекстами

Дано:

- V - словарь слов
- C - множество контекстов

Можем построить матрицу S размера $|V| \times |C|$, элементы которой будут описывать связь слова w_i с контекстом c_j.

Например, можно взять положительную поточечную взаимную информацию (PPMI):

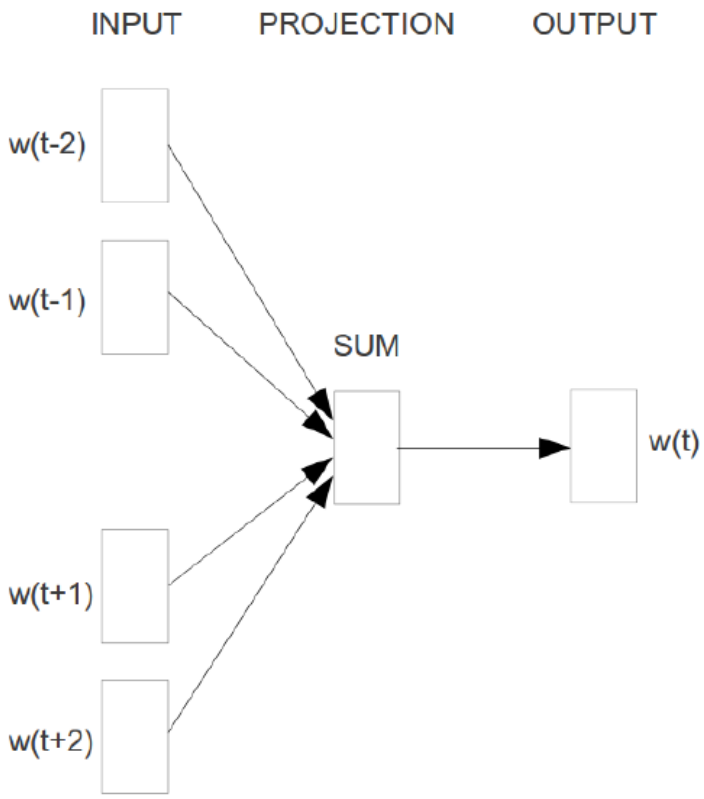
$$S_{i,j} = max(PMI(w_i, c_j), 0),$$

$$PMI(w, c) = log \frac{p(w, c)}{p(w)p(c)} = log \frac{freq(w, c)|V|}{freq(w)freq(c)}$$

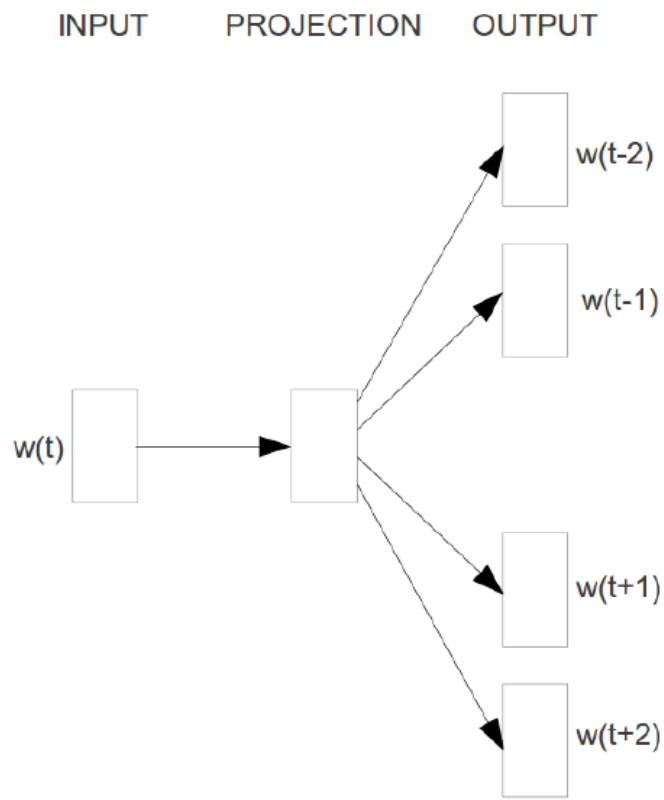
✖ Очень большая размерность матрицы

Word2Vec (2013)

Архитектуры CBOW и Skip-gram

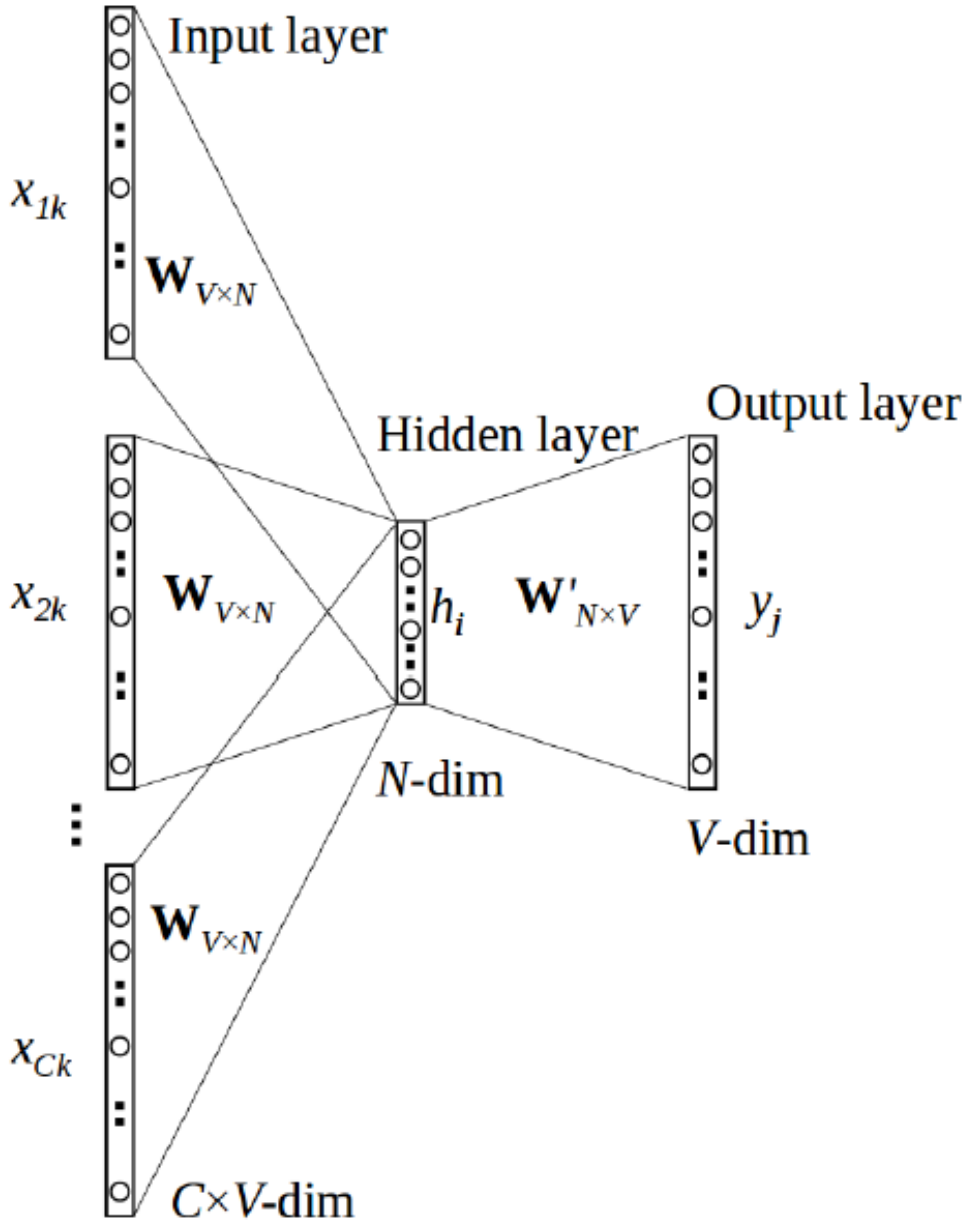


CBOW

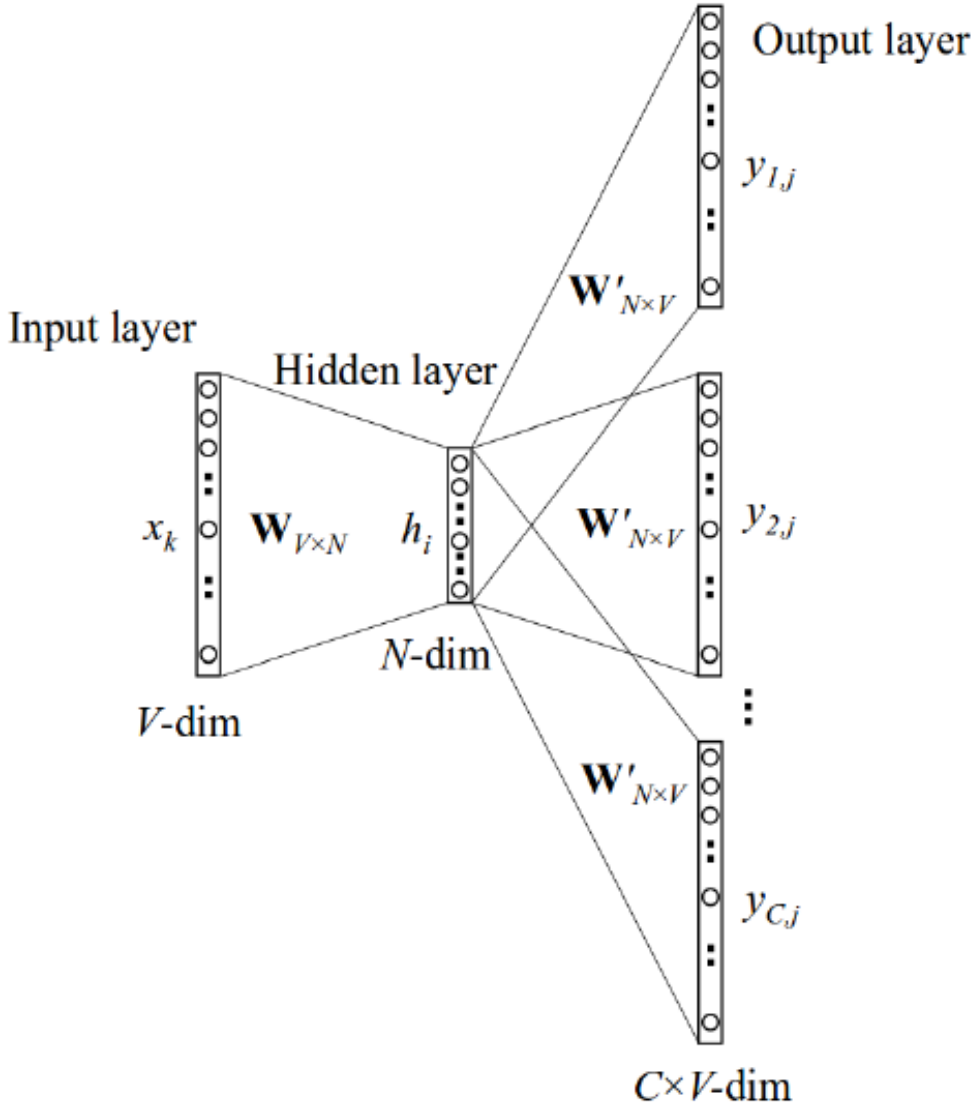


Skip-gram

CBOW (Continious Bag of Words)



Skip-gram


Output layer
$y_{1,j}$
$\mathbf{W}'_{N\times V}$
Input layer
Hidden layer
x_k
$\mathbf{W}_{V\times N}$
h_i
$\mathbf{W}'_{N\times V}$
$y_{2,j}$
N-dim
V-dim
$\mathbf{W}'_{N\times V}$
$y_{C,j}$
$C\times V$-dim

Skip-gram

Оптимизируем

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} log\ p(w_{t+j}|w_t)$$

Вычисление вероятностей выходных слов

Используется softmax

$$p(w_O|w_I) = \frac{\exp\langle \mathbf{v}'_{w_O}, \mathbf{v}_{w_I} \rangle}{\sum_{w=1}^{|V|} exp(\langle \mathbf{v}'_w, \mathbf{v}_{w_I} \rangle)}$$

На практике эту формулу применять сложно, так как вычисление градиента пропорционально $|V|$.

На практике применяют разные аппроксимации: иерархический softmax или negative sampling.

Negative sampling

Идея:

Не будем рассматривать все слова из словаря, а учтем только рассматриваемое слово + подмешаем еще k отрицательных примеров.
Заменим $log\, p(w_O|w_I)$ на

$$log\, \sigma(\langle \mathbf{v}'_{w_O}, \mathbf{v}_{w_I} \rangle) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} [log\, \sigma(-\langle \mathbf{v}'_{w_i}, \mathbf{v}_{w_I} \rangle)]$$

- $k \approx 5 - 20$ для небольших выборок
- $k \approx 2 - 5$ для больших данных

Смысл Negative Sampling

- Фактически, мы имеем два распределения слов: “положительное” (D) и “отрицательное” (N).
- Мы их смешиваем в пропорции $1 : k$
- Задача модели: угадать, из какого распределения пришло слово

Смысл Negative Sampling

По предположению,

$$p(D|w,c) = \sigma(\langle w,c\rangle)$$

Но по формуле Байесса

$$p(D|w,c) = \frac{p(w,c|D)p(D)}{p(w,c|D)p(D) + p(w,c|N)p(N)}$$

Считаем, что контексты в негативных примерах не зависят от слова:

$$p(w,c|N) = p(w|D)p(c|D)$$

Таким образом

$$p(D|w,c) = \frac{p(w,c|D)\frac{1}{k+1}}{p(w,c|D)\frac{1}{k+1} + p(w,c|N)\frac{k}{k+1}} =$$

$$= \frac{1}{1 + k\frac{p(w|D)p(c|D)}{p(w,c|D)}}$$

Смысл Negative Sampling

Заметим, что выражение стоящее в знаметеле очень похоже на взаимную информацию

$$PMI(w, c) = log \frac{p(w, c|D)}{p(w|D)p(c|D)}$$

Таким образом,

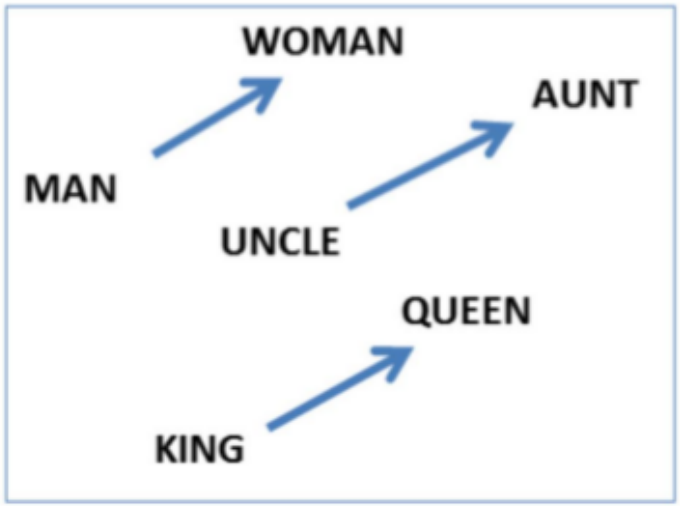
$$p(D|w, c) = \frac{1}{1 + ke^{-PMI(w,c)}}$$

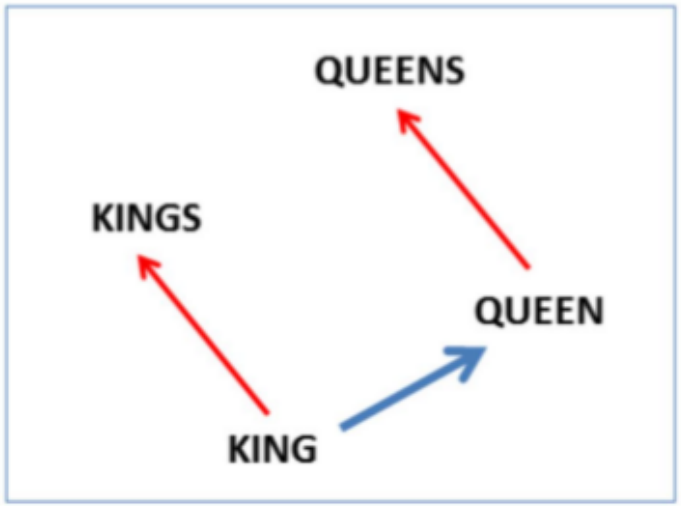
$$\langle w, c \rangle = PMI(w, c) - ln\, k - \text{“сдвинутый” } PMI$$

Skip-gram Negative Sampling эквивалентен факторизации матрицы “сдвинутого” PMI.

Свойства выученных представлений

$v(\text{king}) - v(\text{man}) + v(\text{woman}) \approx v(\text{queen})$

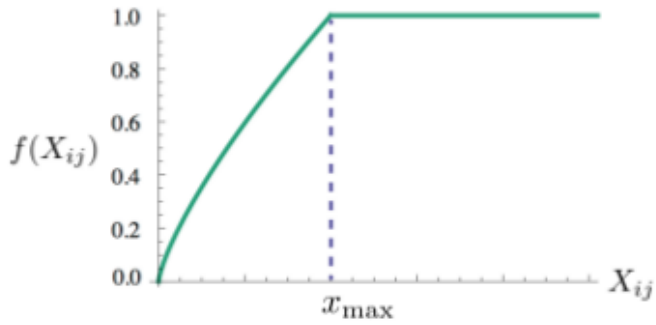




GLoVe (2014)

$$J(\theta) = \frac{1}{2} \sum_{i,j=1}^{W} f(P_{ij})(u_i^T v_j - \log P_{ij})^2$$

$$X_{final} = U + V$$



Word2vec

Вопрос

- Как это использовать в поиске?

Вопросы