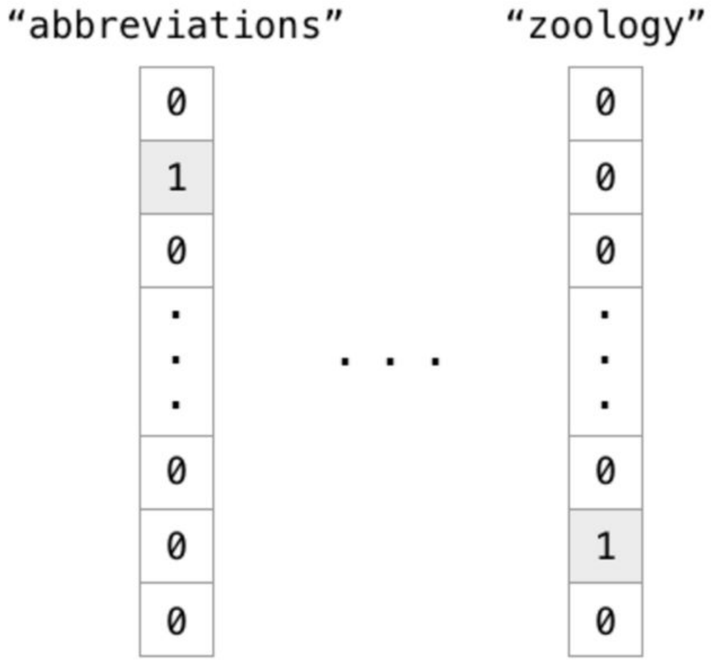
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Компьютерная лингвистика и информационные технологии

Неделя 8: Векторные представления, часть 1
(на основе лекций М. Апишева и Е. Артемовой, ФКН ВШЭ)

One-hot, счетные, сингулярные векторы

- Разреженные, ортогональные векторные представления
- Размерность представлений не фиксирована
- При появлении новых объектов необходимо перестраивать признаковую матрицу
- Нерепрезентативное кодирование порядка слов
- Семантика объектов не кодируется
- Отношения между объектами текста не кодируются



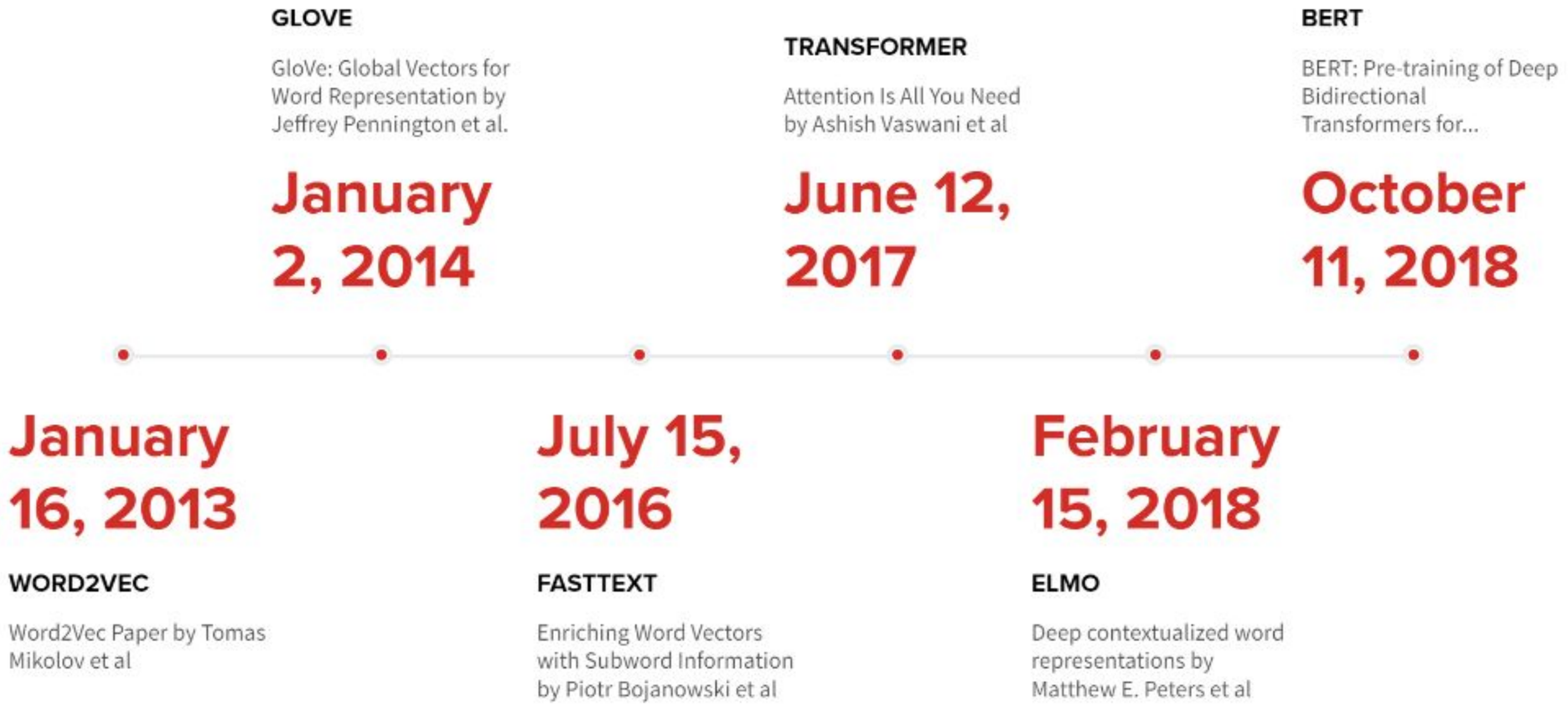
Векторные представления*

- Векторное представление (embedding) – сопоставление произвольному объекту некоторого числового вектора в пространстве фиксированной размерности
- Вход – коллекция документов *D*
- Выход – векторные представления слов из словаря *V*, построенного по коллекции документов *D*

*word vector, word representation, word embedding, word vector representation

$$banking = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \end{pmatrix}$$

Векторные представления



Расстояние между векторами

d(x, y), где x и y – векторы.

- Манхэттенское расстояние
- Евклидово расстояние
- Косинусное расстояние

Семантические ассоциаты для *разум* (ALL)

Частотность слова

☑ Высокая ☑ Средняя ☐ Низкая

НКРЯ и Wikipedia

1. рассудок NOUN 0.78
2. ум NOUN 0.65 
3. здравый ADJ 0.64
4. разум PROPN 0.62 
5. разумение NOUN 0.62
6. сознание NOUN 0.61 
7. истина NOUN 0.61 
8. познание NOUN 0.60
9. мудрость NOUN 0.59 
10. мышление NOUN 0.59

Манхэттенское расстояние
Manhattan or cityblock distance

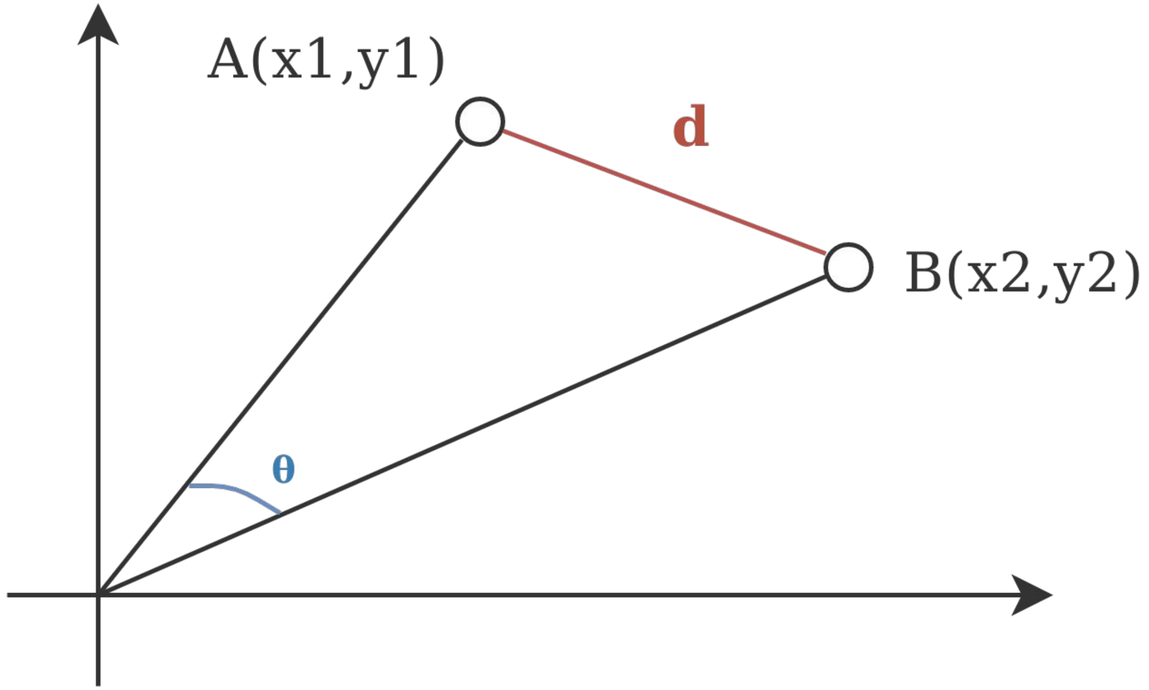
$$d_1(\mathbf{p}, \mathbf{q}) = \|\mathbf{p} - \mathbf{q}\|_1 = \sum_{i=1}^{n} |p_i - q_i|$$

Евклидово расстояние
Euclidean distance

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

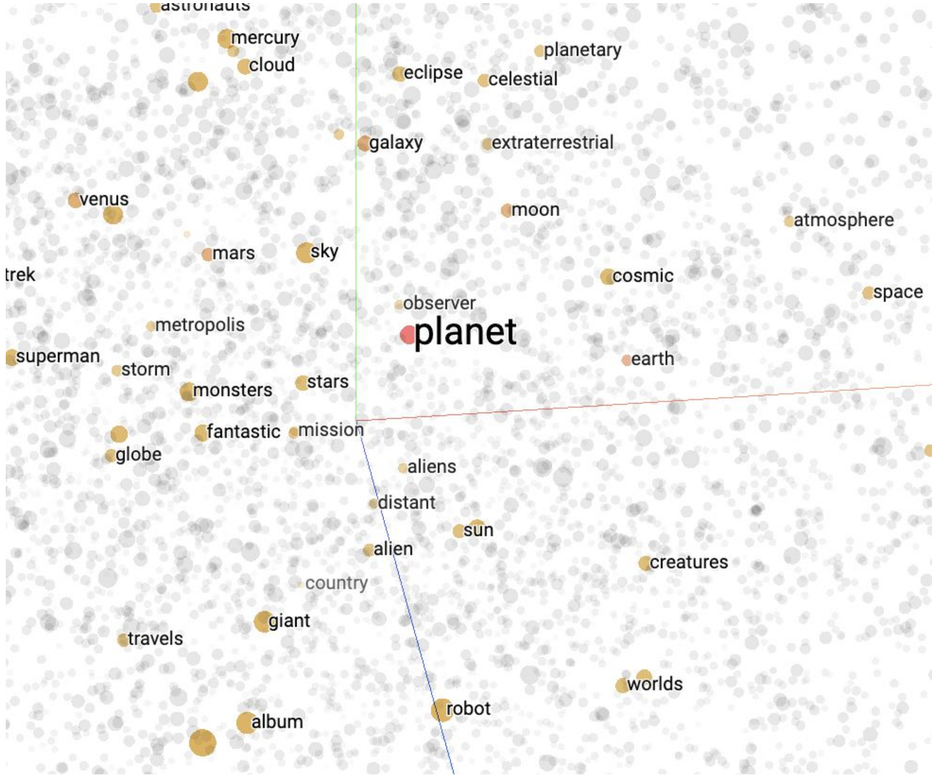
Косинусное расстояние
Cosine similarity

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}},$$

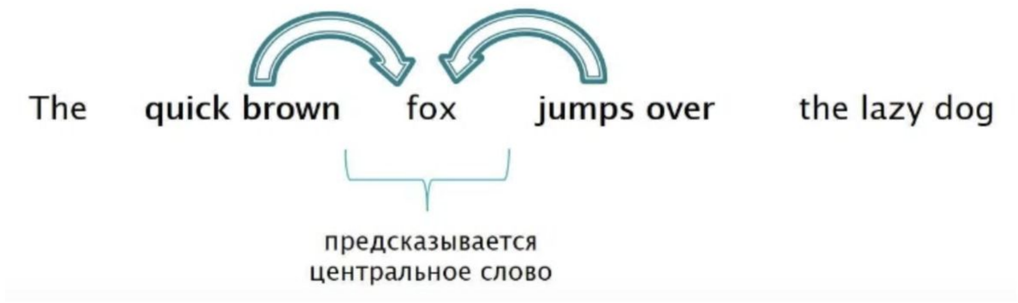


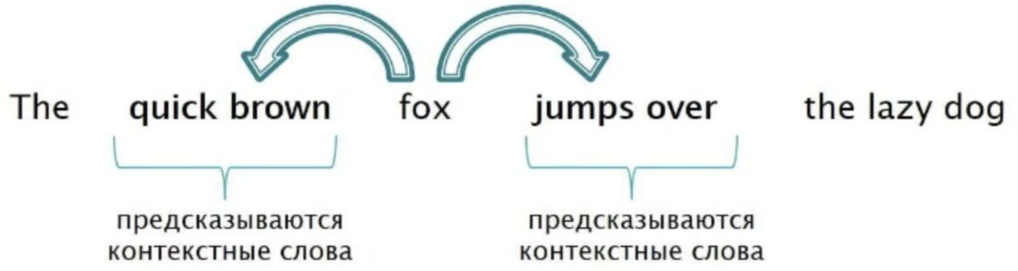
word2vec

- Дистрибутивная гипотеза
- Полученные представления называются *дистрибутивными*
- Получение векторных представлений слов в пространстве невысокой фиксированной размерности
- Близкие по смыслу слова встречаются в похожем контексте –> рядом в векторном пространстве



Формулировка задачи





Continuous Bag of Words (CBOW)

Skip-gram

Softmax

- Софтмакс позволяет преобразовать произвольные значения в вероятностное распределение
- Получаемые значения находятся в диапазоне [0; 1], и их сумма равна 1

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

σ = softmax

$\vec{z}$ = input vector

e^{z_i} = standard exponential function for input vector

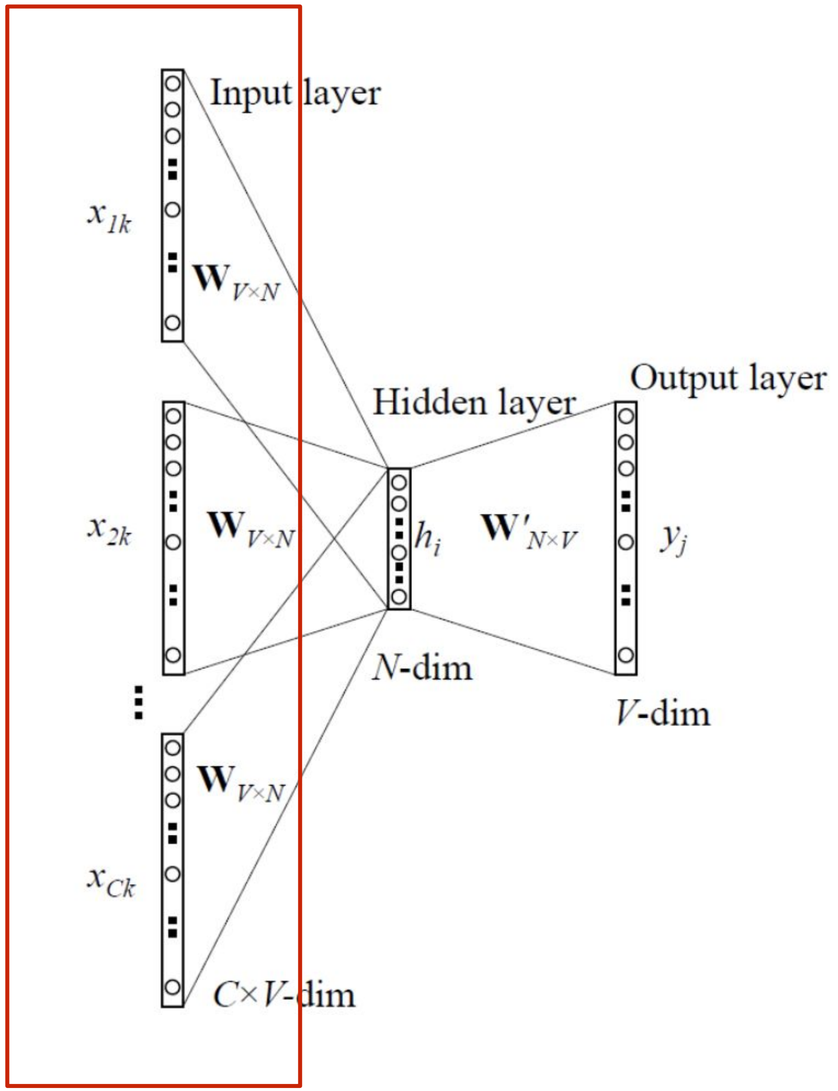
K = number of classes in the multi-class classifier

e^{z_j} = standard exponential function for output vector

CBOW

- Данные: контексты размера 2C + 1
- Вход: one-hot векторы для каждого слова в контексте
- Получаем матрицу объектов-признаков (2C, vocab_size)

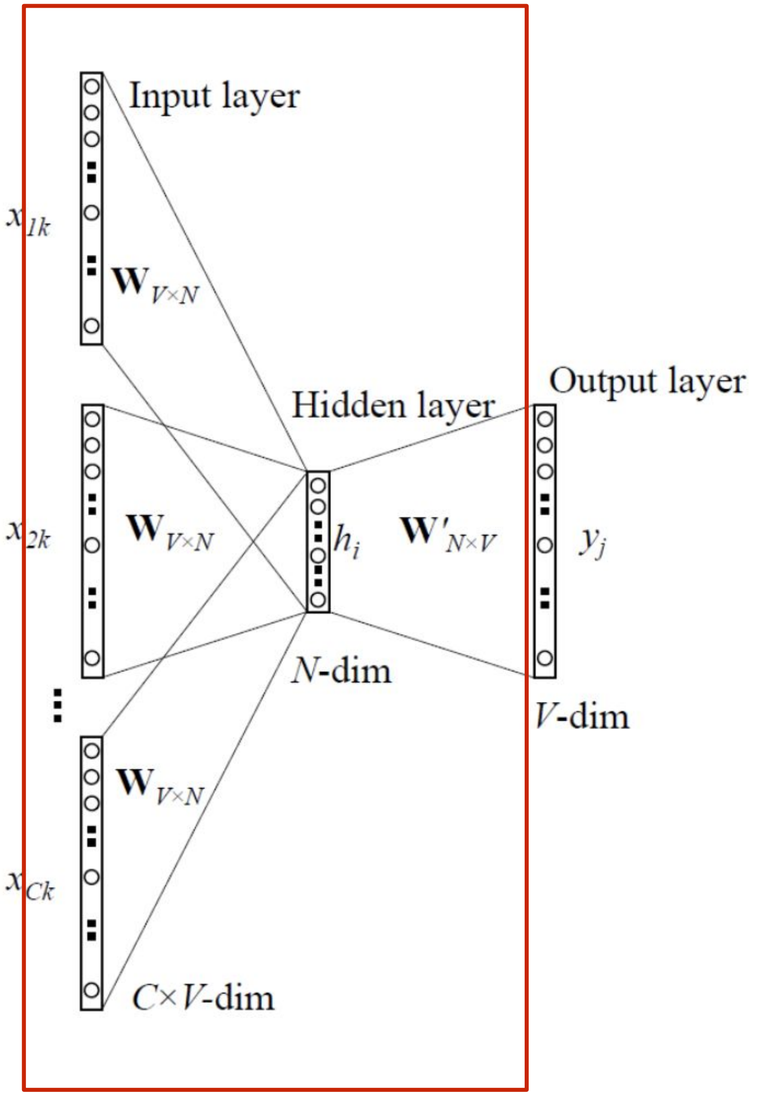
мама ? раму



CBOW

- Имеем скрытое состояние W размерности (vocab_size, embedding_dim)
- Применяем преобразование, умножая признаковую матрицу (2C, vocab_size) на матрицу W
- Получаем матрицу контекста (2C, embedding_dim)
- Усредняем вектор скрытого состояния для контекста:

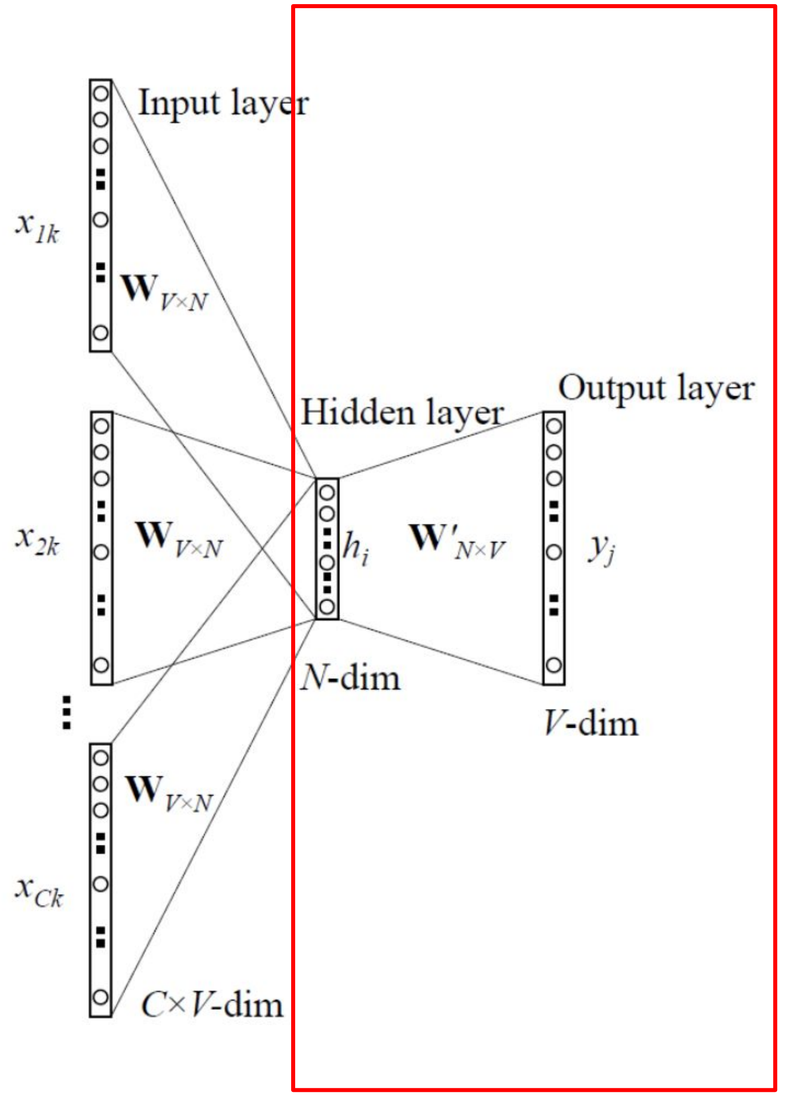
мама ? рамку



CBOW

- Умножим вектор скрытого состояния для контекста на вторую матрицу весов W’ (embedding_dim, vocab_size)
- Получаем вектор (1, vocab_size)
- Берем софтмакс по размеру словаря

мама ? рамку

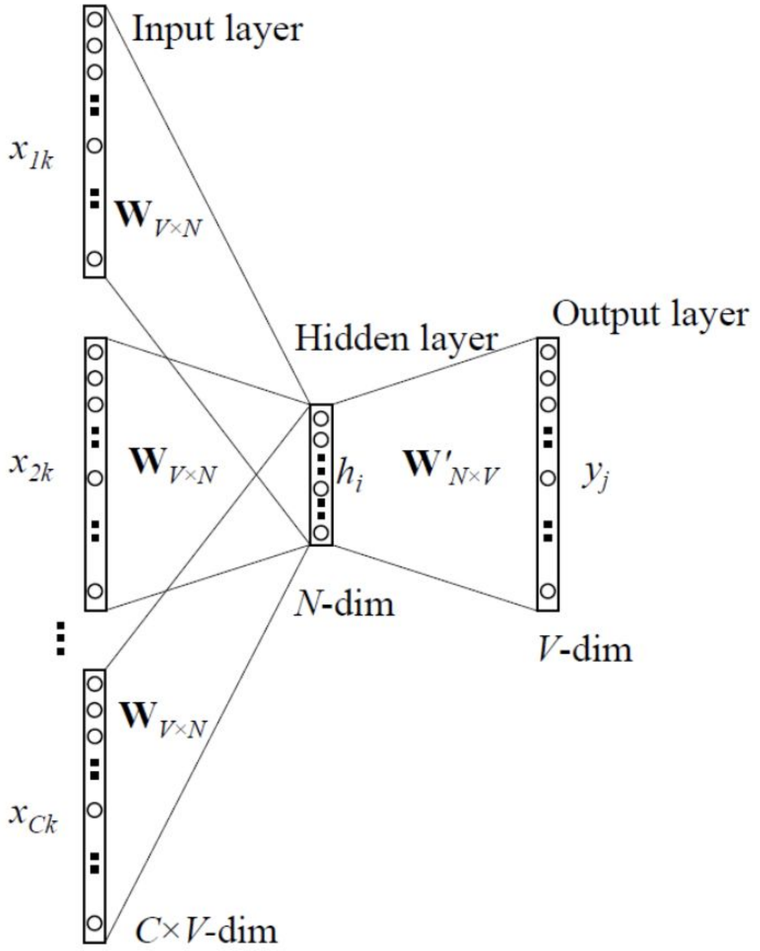


мыла

CBOW

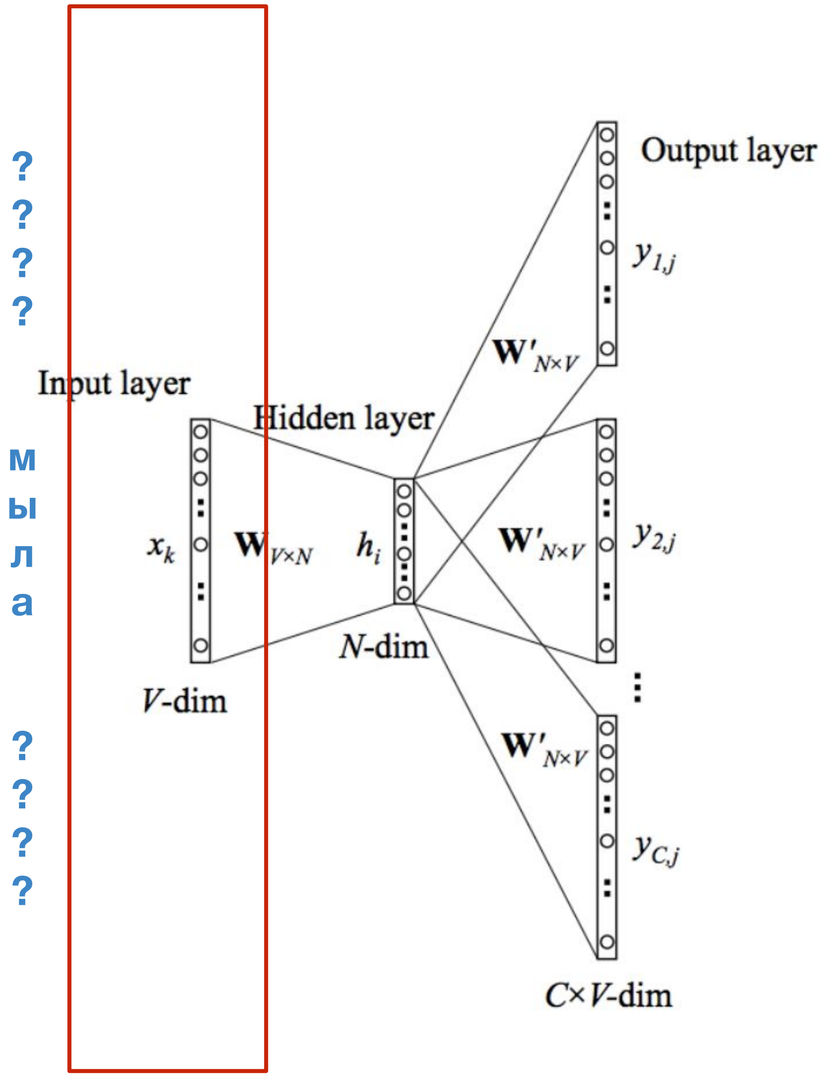
- Получаем две матрицы W и W’
- В них одинаковые слова, но разные векторы.

  Почему?
- В качестве эмбеддингов используется матрица W
- Подробнее тут (4.2 CBOW)



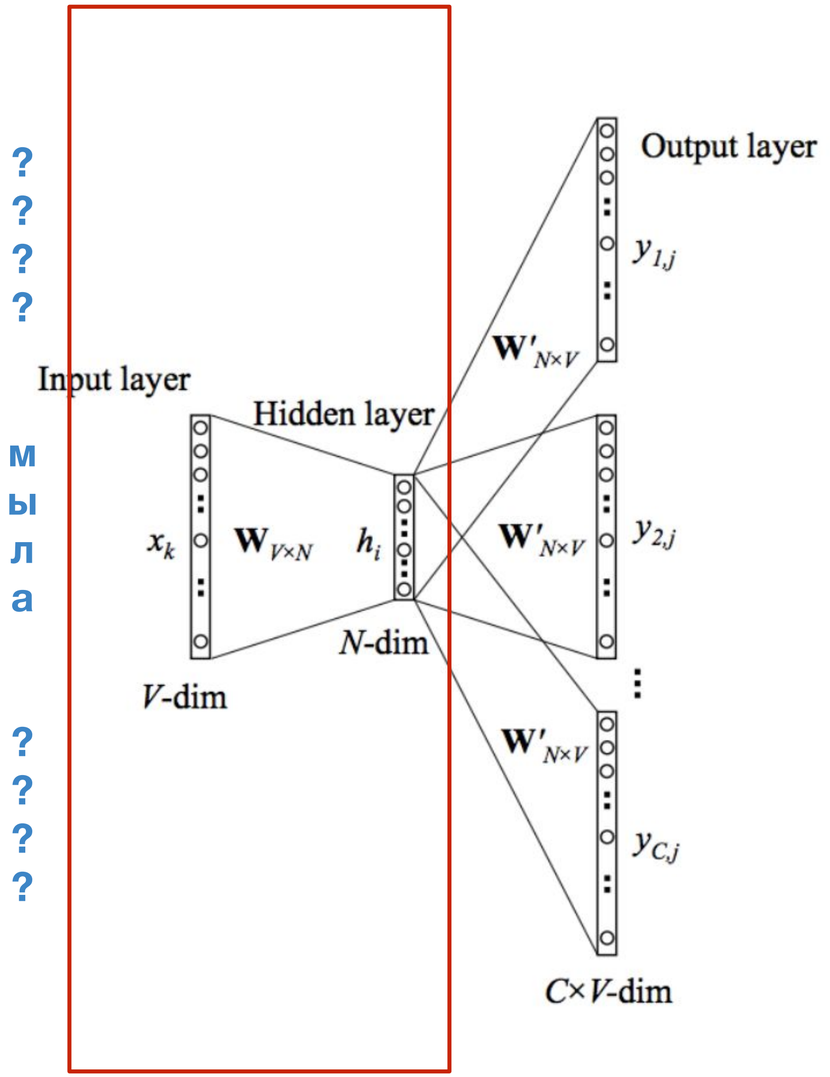
Skip-gram

- Данные: контексты размера 2C + 1
- Мы все равно не кодируем информацию о порядке слов, о расстоянии слов до центрального – важен контекст
- Вход: one-hot вектор центрального слова



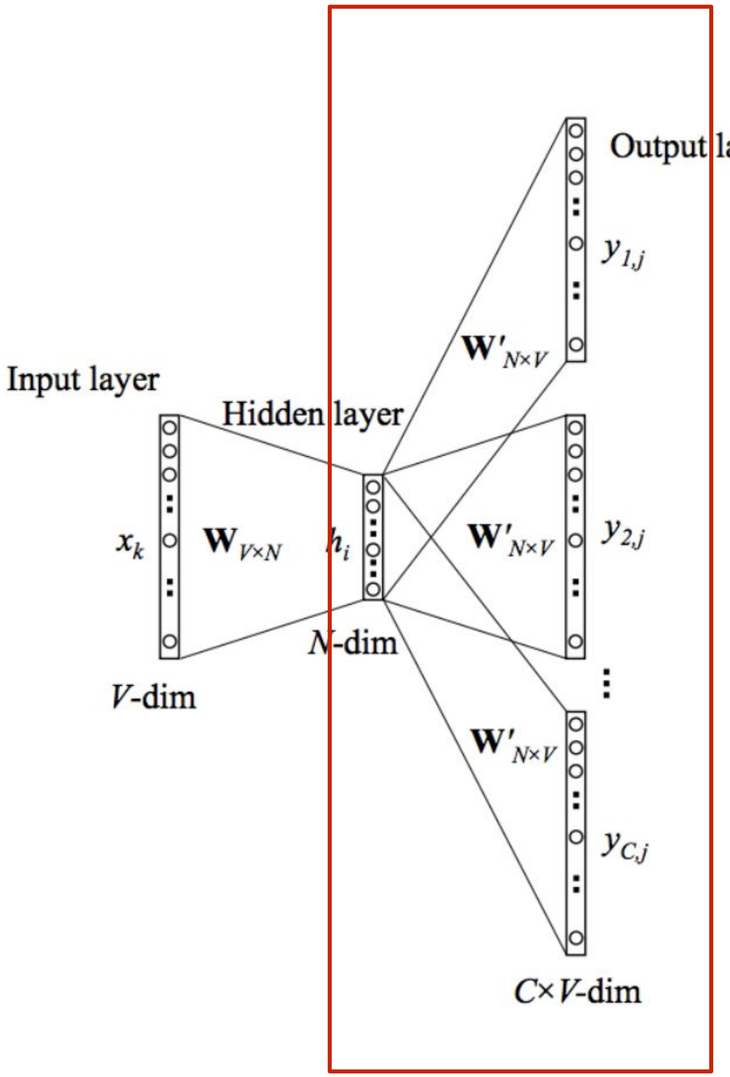
Skip-gram

- Имеем скрытое состояние W размерности (vocab_size, embedding_dim)
- Применяем преобразование, умножая one-hot вектор (1, vocab_size) на матрицу W
- Получаем вектор скрытого состояния и используем его, чтобы предсказать слова в контексте



Skip-gram

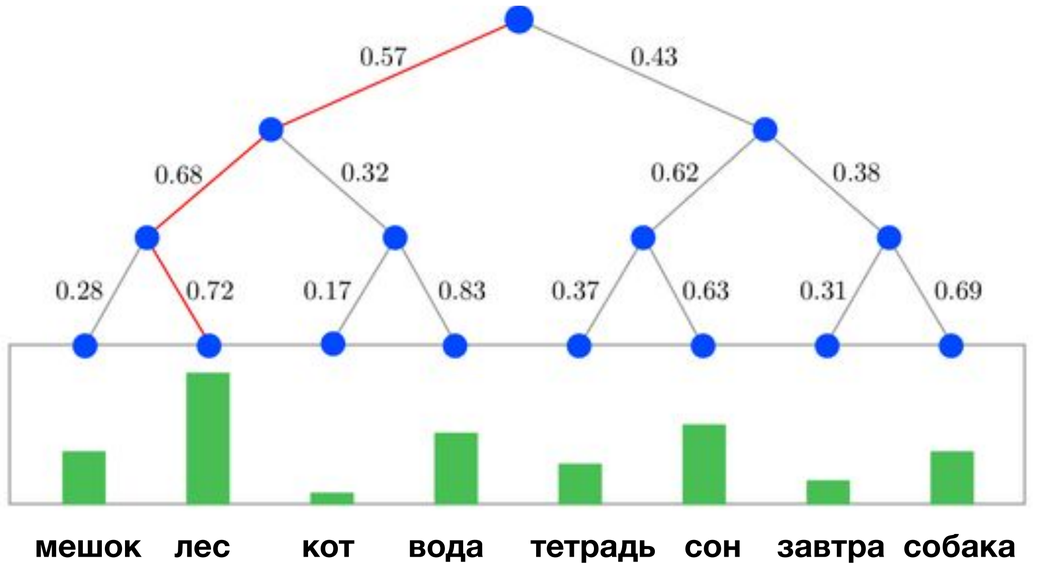
- Применяем преобразование, умножая one-hot вектор (1, vocab_size) на матрицу W
- Получаем вектор скрытого состояния (1, embedding_dim)
- Умножаем на вторую матрицу W’ (embedding_dim, vocab_size)
- Применяя софтмакс, получаем вектор вероятностей, посчитанных независимо для каждого слова в контексте



мама рамку

Оптимизация

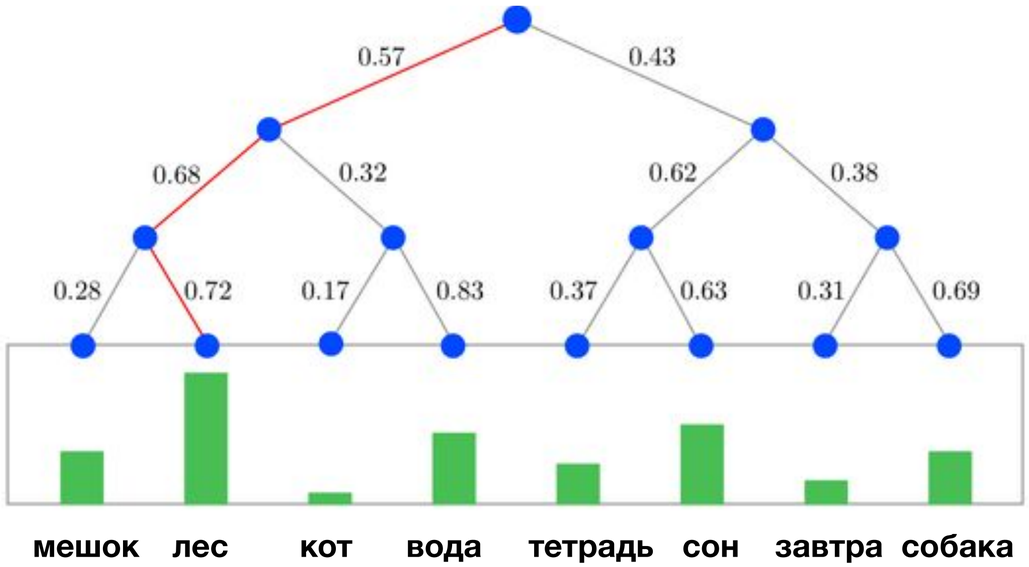
- Софтмакс – дорогое удовольствие O(n)
- Методы оптимизации:
  - Иерархический софтмакс
  - Skip-gram with Negative Sampling



Оптимизация

- Итоговая вероятность спуска к целевому листу “лес”:

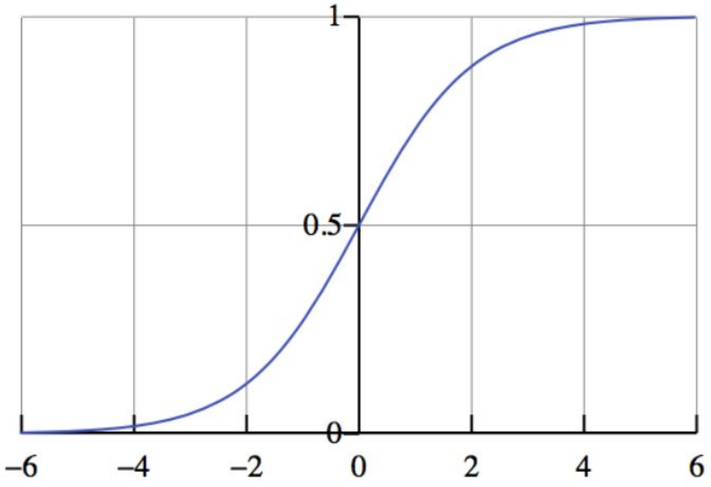
  p(лес | hidden) = 0.57 * 0.68 * 0.72 = 0.28
- Из вектора вероятностей используем только 2C компонент
- Сложность спуска по дереву к целевому листу **O(nlogn)**



Negative Sampling (SGNS)

- Отказ от дорогих удовольствий
- Поставим задачу бинарной классификации
- Рассмотрим пару слов (w1, w2):
  - Класс 0: w1 не входит в контекст w2
  - Класс 1: w1 входит в контекст w2
- Используем сигмоиду, чтобы оценить вероятность того, что два слова встретились рядом

The **sigmoid** function
$\sigma(x) = \frac{1}{1+e^{-x}}$
is the 1D version of the softmax and can be used to model a probability



$$p(z = 1|(w, s)) = \frac{1}{1 + \exp(-v_w^T v_s)} = \sigma(v_w^T v_s)$$

Negative Sampling (SGNS)

- Имеем примеры класса 1, но нет примеров класса 0
- Выбираем ненаблюдаемые пары слов на каждой итерации обучения
- Примеров класса 1 существенно меньше, чем примеров класса 0
- Из множества ненаблюдаемых пар слов D2 семплируем фиксированное количество пар

$$\mathcal{L} = \sum_{(w,s)\in D_1} \log \sigma(v_w^T v_s) + \sum_{(w,s)\in D_2} \log \sigma(-v_w^T v_s),$$

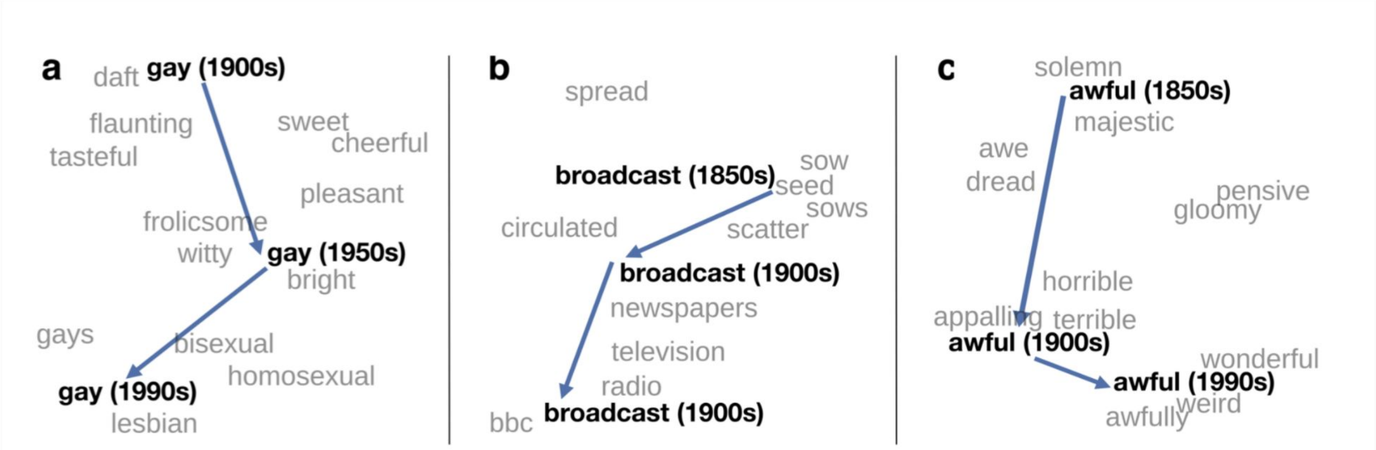
$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}}$$

Сделаем выводы: word2vec

- Одно слово – один вектор
- Не требуется разметка, но можем использовать частеречную разметку: **стекло_NOUN**
- Решаем многоклассовую классификацию, где количество классов равно мощности словаря
- Out-of-vocabulary слова
- Тексты могут быть представлены:
  - усреднением векторов слов
  - суммой векторов слов
  - конкатенацией векторов слов

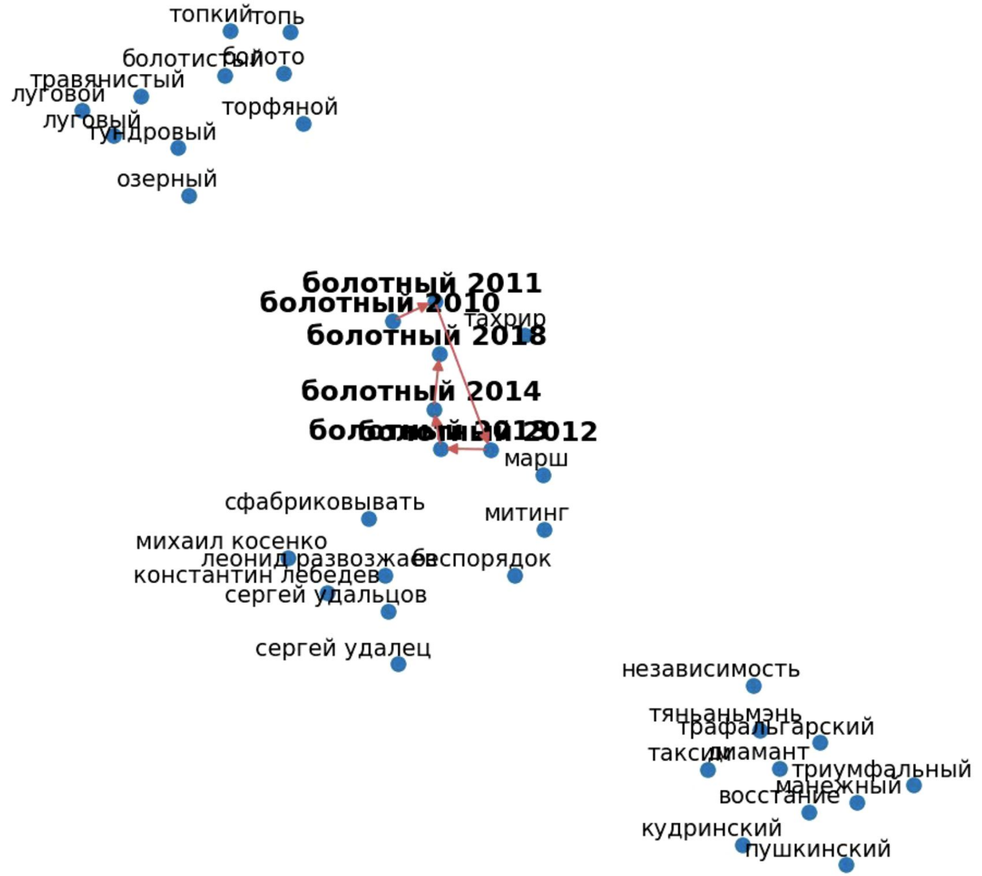
Определение семантического сдвига

- https://shiftry.rusvectores.org/ru/
- Дано k коллекций документов, принадлежащих разным временным периодам
- Обучим модели за каждый период
- Прокрустово выравнивание векторов в одно пространство



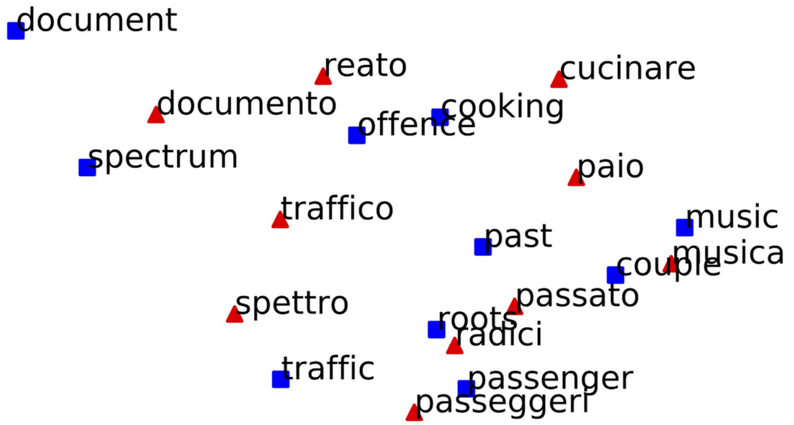
Определение семантического сдвига

- Косинусное расстояние (w1, w2)
- Список семантических ассоциатов
- Контексты для w1 и w2



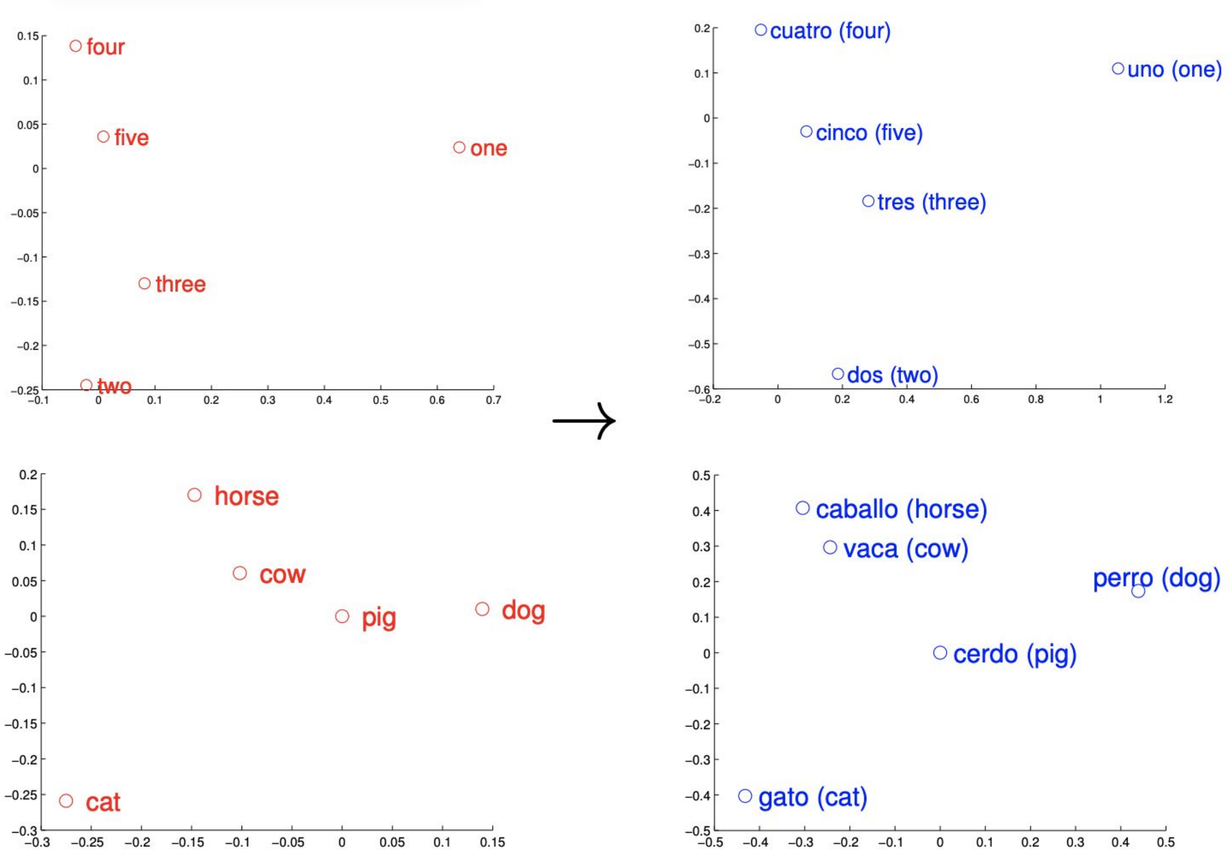
Мультиязычные векторные представления

- **Моноязычный**: обучаем эмбеддинги для каждого языка отдельно и линейные преобразования между моноязычными пространствами
- **Псевдо-мультиязычный**: обучаем эмбеддинги одновременно на синтетическом корпусе, в котором перемешаны слова на разных языках
- **Мультиязычный**: обучаемся на параллельном корпусе или на конкатенированных моноязычных коллекциях текстов



Exploiting Similarities among Languages for MT

- (Mikolov et al., 2013)
- Близкие слова на разных языках образуют семантические кластеры
- Можем выучить линейное преобразование между двумя пространствами
- k наиболее частотных слов и их переводы
- Linear projection + SGD



Random Translation Replacement

- (Gouws and Sogaard, 2015)
- Автоматически переводим коллекцию документов
- С вероятностью p заменяем слова на перевод
- Обучаем CBOW

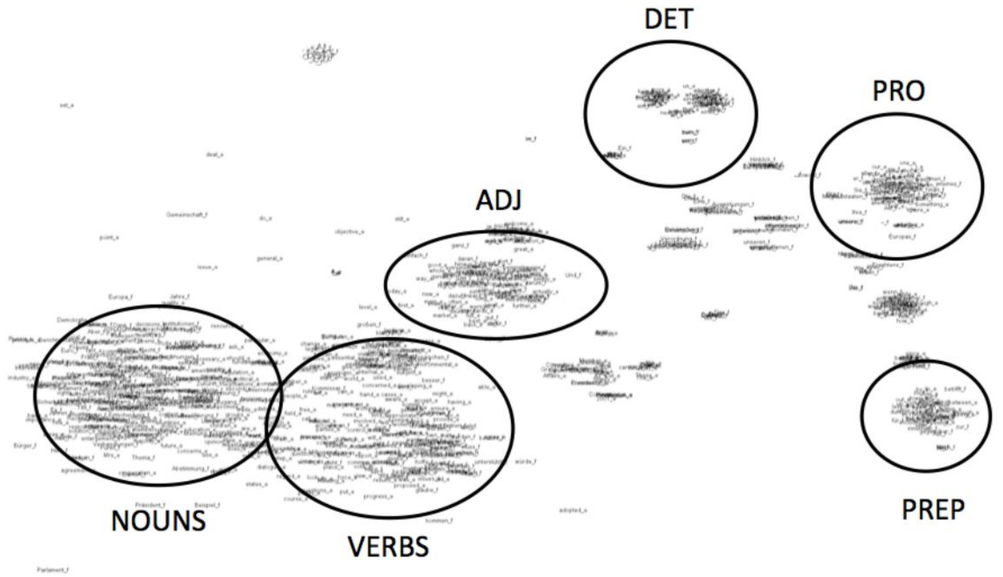
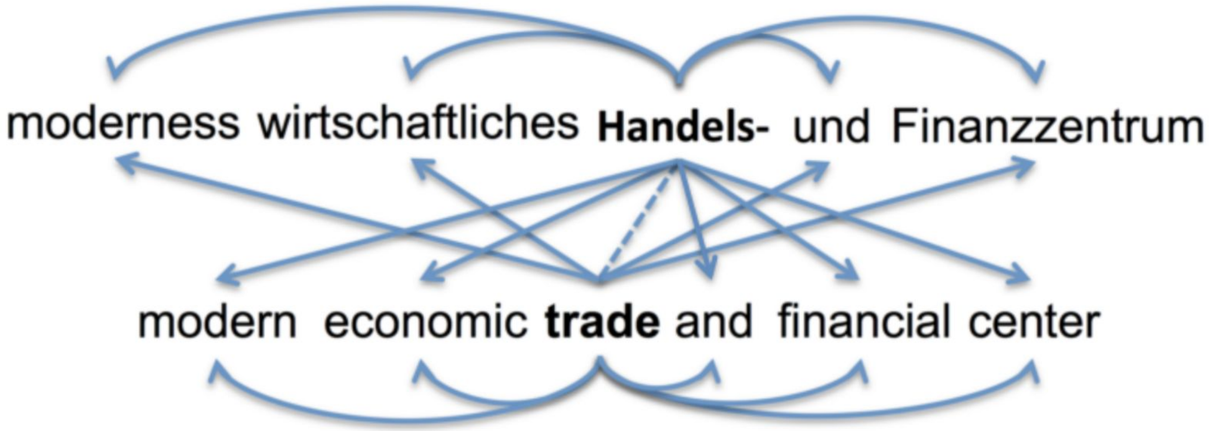


Figure 1: t-SNE visualization of BARISTA embeddings trained with POS classes.

Bilingual Skip-gram

- (Luong et al., 2015)
- Нужны выровненные параллельные предложения



Полезные материалы

- (Mikolov et al., 2013)
- SGNS (Levy and Goldberg, 2014)
- CS224N: лекция
- CS224N: заметки
- Лекция Мурата Апишева
- Визуализация векторных представлений
- Обзор по мультиязычным эмбеддингам