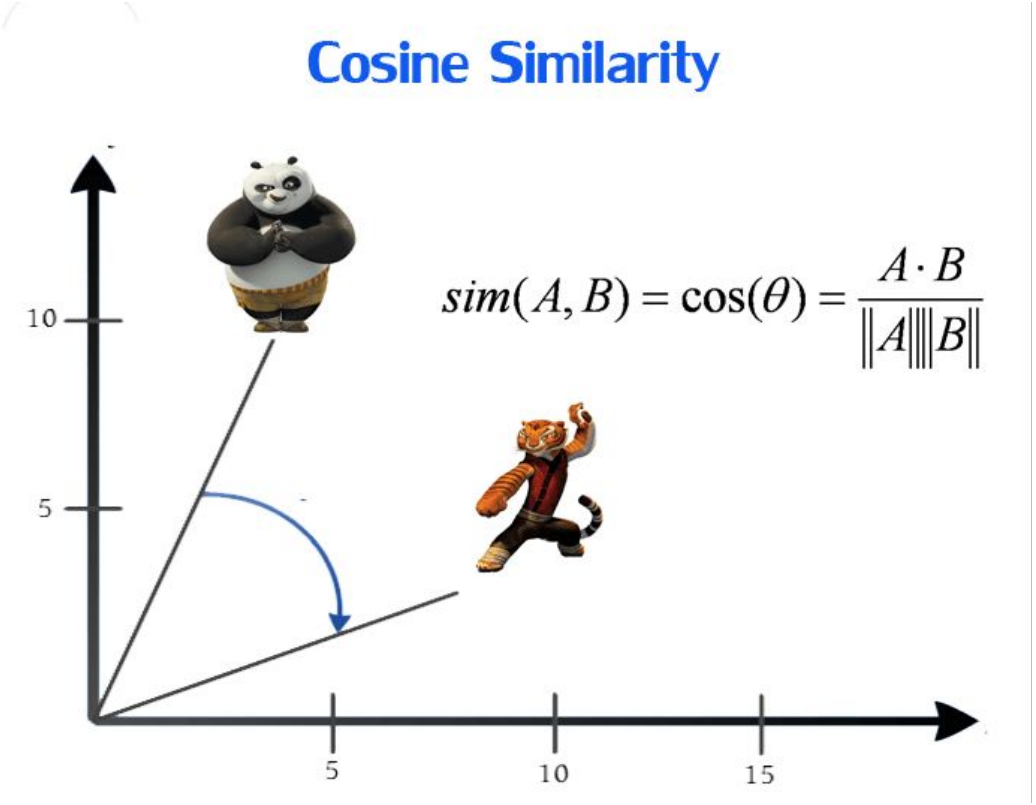
Эмбеддинги слов. WMD.

Маша Шеянова, masha.shejanova@gmail.com

Эмбеддинги

Как найти, насколько близки слова?



- надо найти способ превратить слова в вектора так, чтобы они отражали **контекст**
- найти расстояние между этими векторами одним из способов

Источник картинки.

Как сделать из слов вектора?

Итак, основная идея — **учитывать контекст**. Но как? Про это есть большая наука.

Самый простой-наивный метод — **счётный**. Идея: для каждого слова возьмём ближайшие в некотором окне (например, -5 +5). Сделаем такой же мешок слов, как делали для документов (CountVectorizer, TfidfVectorizer). Можно делать “скользящее окно”.

Плюсы: легко и быстро.

Минусы: для большого корпуса — очень большие вектора.

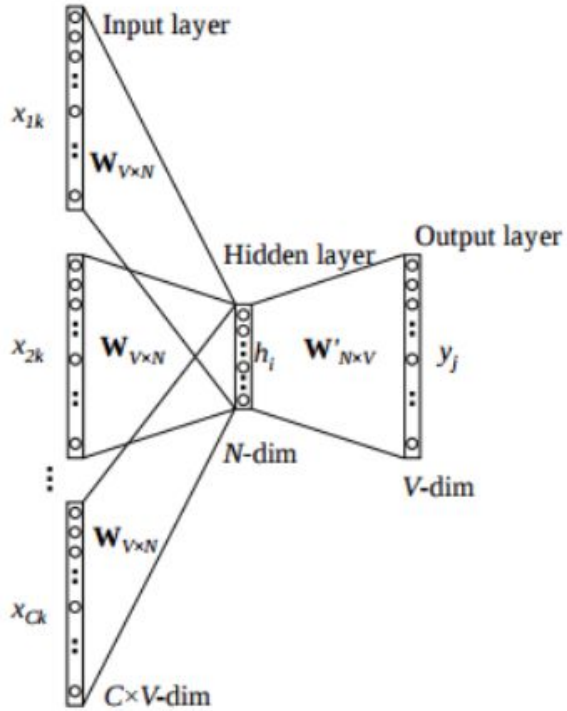
Word2vec

В двух словах, Word2Vec — это метод строить гораздо более компактные эмбеддинги с помощью нейросетей.

Методы:

- CBOW (Common Bag Of Words)
- skipgram

CBOW (common bag of words)



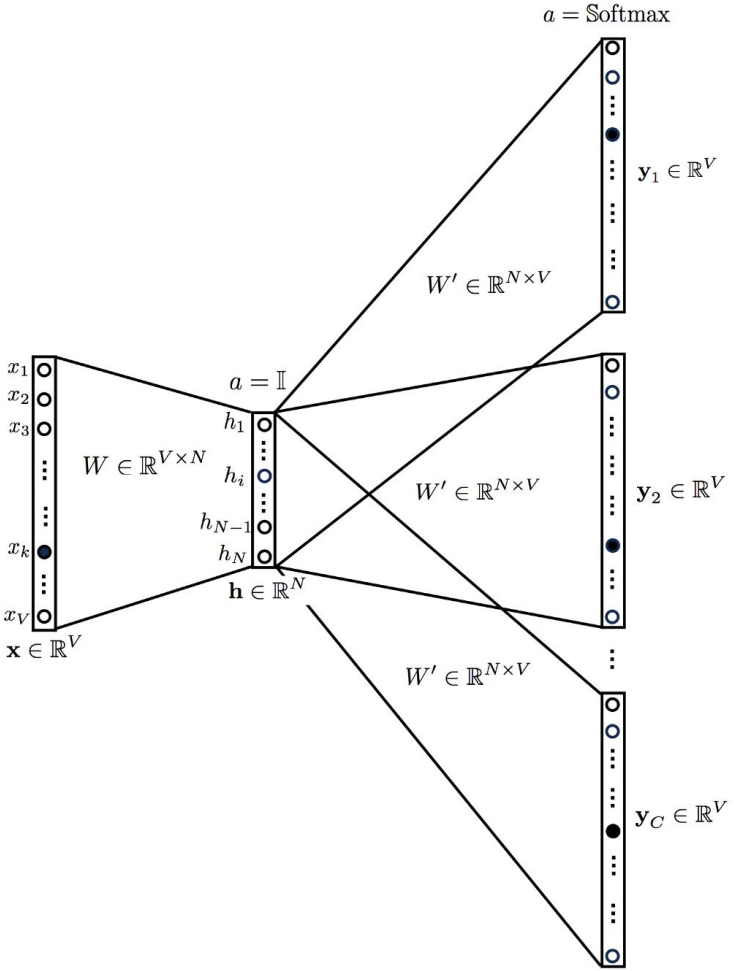
Источник картинки

Метод CBOW пытается **предсказать слово по его контексту**. Он берёт каждое слово из контекста слова Y и пытается по нему предсказать слово Y.

skipgram

skipgram, в отличие от CBOW, пытается предсказывать контекст по слову.

- **Skip Gram** хорошо работает с маленьким объёмом данных и **лучше представляет редкие слова**
- **CBOW** работает быстрее и **лучше представляет наиболее частые слова**



Веб-интерфейсы и ресурсы про word2vec

rusvectores — для русского

tutorial по word2vec — для английского

хорошее объяснение про word2vec и fasttext (англ)

word2vec tutorial на kaggle

Fasttext

Fasttext — почтиии то же самое, что и word2vec, но работает на уровне меньше, чем слово.

Идея такая: разбиваем каждое слово на *символьные нграммы*. Например, так: **apple → app**, **ppl**, **ple**

Обучаем нейросетку так, чтобы получить эмбеддинги этих кусочков. Финальный эмбеддинг слова — сумма эмбеддингов его кусочков.

В чём профит? Умеем представлять даже слова, которых не было в корпусе!

GloVe

- идея окна-контекста, как в Word2Vec
- вместо слов, предсказываем соотношения вероятностей совстречаемости слов

$$F(w_i, w_j, \tilde{w}_k) \approx \frac{P_{ij}}{P_{jk}}$$

$$P_{ij} = \frac{\text{number of times j appeared in context of i}}{\text{number of words that appeared in context of i}}$$

- тем самым, используем “глобальную информацию” о совстречаемости по всему корпусу

Где взять готовые эмбеддинги

Можно обучить свои эмбеддинги. Но это долго и не всегда нужно. Есть ли уже обученные эмбеддинги? Конечно!

Rusvectores! (для русских слов)

Как заэмбеддить текст

Простые способы

- сумма эмбеддингов слов
  - информация о важности теряется
- сумма с tfidf весами
  - лучше, но вычислительно сложнее и не всегда работает
- ключевые слова, термины
  - например, взять только их
  - или придать им больший вес

WMD

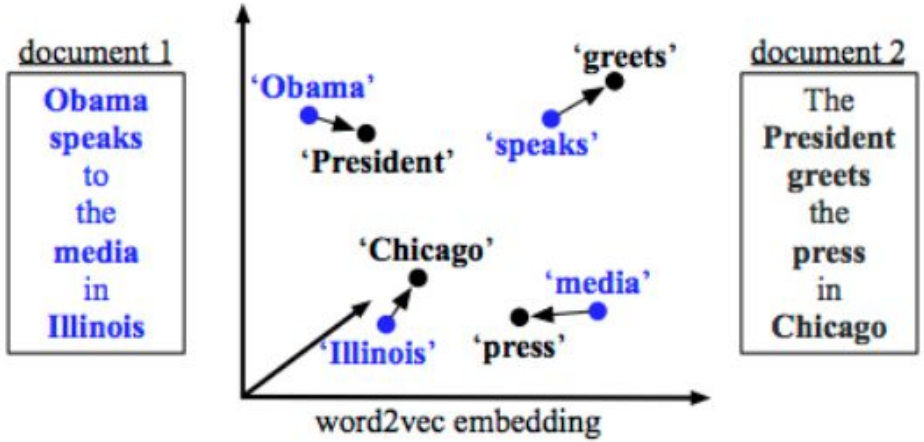


Figure 1. An illustration of the *word mover's distance*. All non-stop words (***bold***) of both documents are embedded into a *word2vec* space. The distance between the two documents is the minimum cumulative distance that all words in document 1 need to travel to exactly match document 2. (Best viewed in color.)

Расстояние между документами — сумма расстояний между ближайшими словами.