Word2vec

Маша Шеянова, masha.shejanova@gmail.com

Эмбеддинги слов

Дистрибутивная семантика

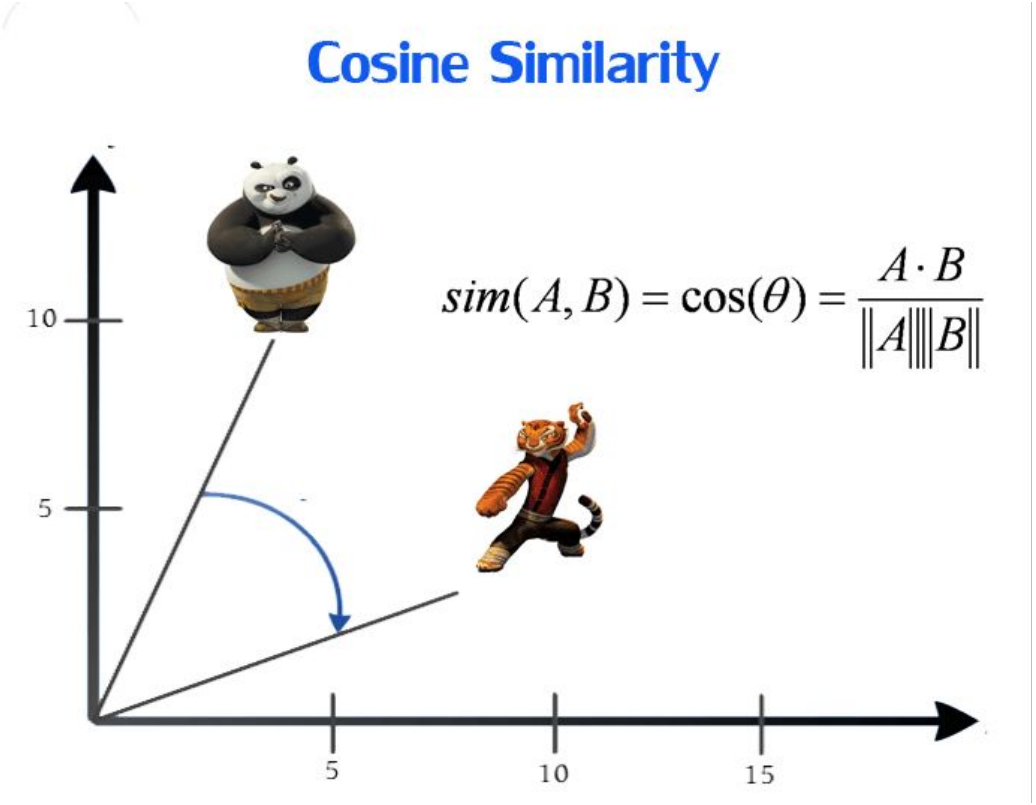
Что мы хотим:

- формальный способ считать лексическую близость
- глобально: научить компьютер извлекать смыслы из текста

Как делать это автоматически?

Дистрибутивная гипотеза: значения слов полностью определяются их контекстами. Слова с похожими типичными контекстами имеют схожее значение.

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? А вот 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 с *CountVectorizer*, но для контекста.

Плюсы: легко и быстро.

Минусы: для большого корпуса — очень большие вектора.

Пример таблички с контекстами

По итогу у овощей будут контексты, похожие друг на друга, а у животных — друг на друга.

	редис	картошка	кот	...	собака
редис	-	5	1		0
картошка	5	-	0		1
кот	1	0	-		6
...					
собака	0	1	6		-

Word2vec

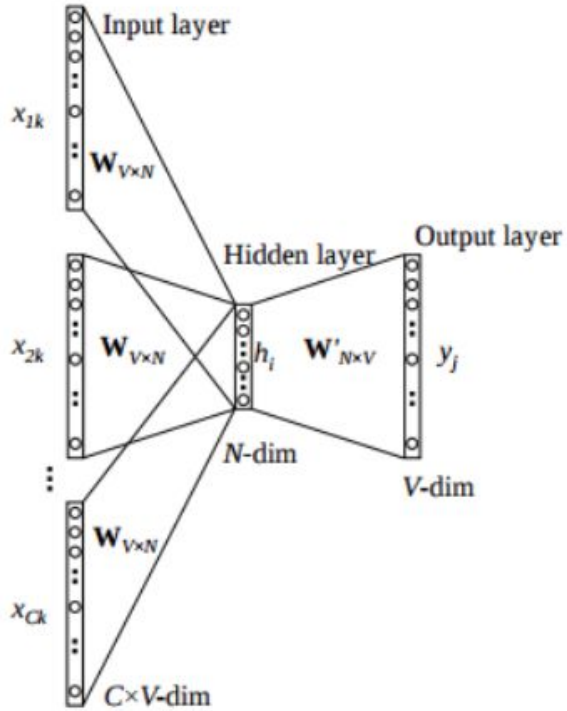
Word2vec

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

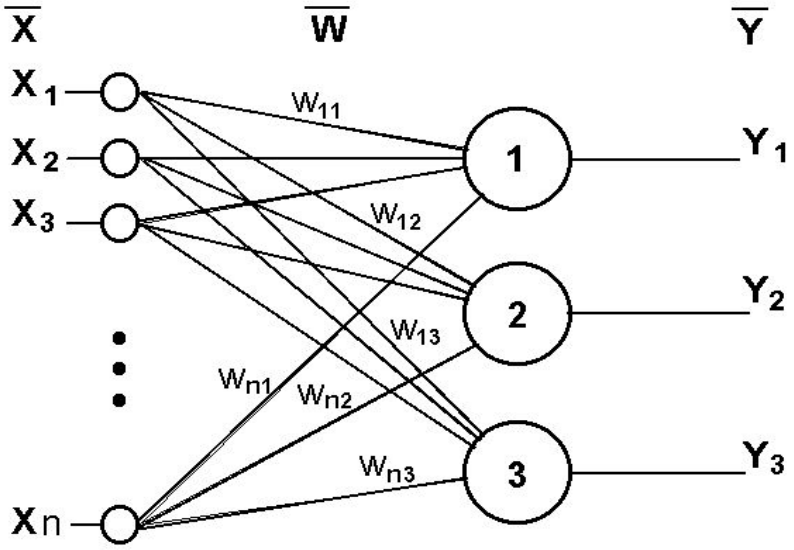
Метод CBOW пытается **предсказать слово по его контексту**. Он берёт каждое слово из контекста слова *y* и пытается по нему предсказать слово *y*.

Здесь, каждый x и каждый y — one-hot вектора, где нули везде, кроме позиции, отвечающей за слово.

А *h* — внутренний слой такой ширины, как мы захотим.

Откуда берутся эмбеддинги

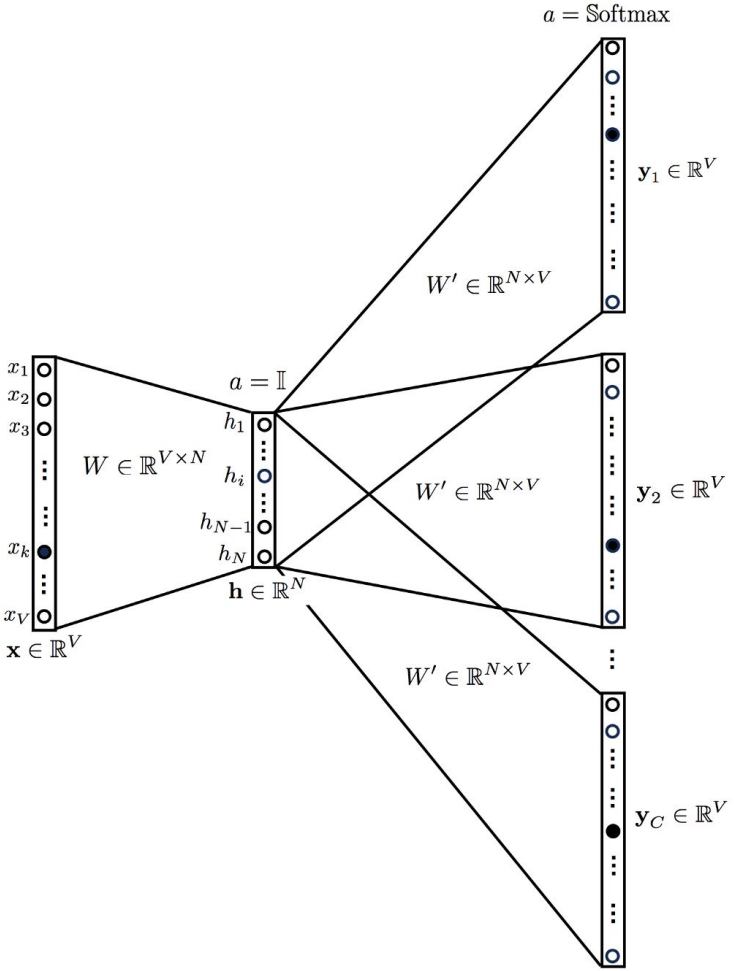
- на картинке — однослойная нейросеть, как word2vec
- веса на стрелочках обучаются (градиентным спуском) так, чтобы слова из контекста хорошо предсказывались
- каждое слово — one-hot вектор (везд нули, кроме позиции слова)
- а значит, информацию о контексте слова номер *k* хранят все веса на стрелочках, выходящих из узла x_k



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**



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!

Byte Pair Encoding (BPE)

Идея: разбивать текст на меньшие единицы, чем слова, но делать это умно, учитывая **частоту совместной встречаемости**.

Алгоритм:

1) вначале, “юнит” — каждая отдельная буква
2) считаем, какая пара юнитов встречается вместе чаще остальных
3) сливаем такую пару, образуя новый юнит
4) повторяем шаги 2-3, пока не достигнем словаря желаемого размера

В результате — у нас есть юниты разного размера, от слова, до морфемы и, наконец, отдельной буквы. И на такой токенизации можно обучать word2vec.

Главное достоинство word2vec

Так как мы можем регулировать ширину внутреннего слоя:

- мы можем “ужать” информацию о контексте слова до его размера, эффективно используя память
- мы можем выбирать силу сжатия
- модели, обученные на больших корпусах не будут весить сильно больше

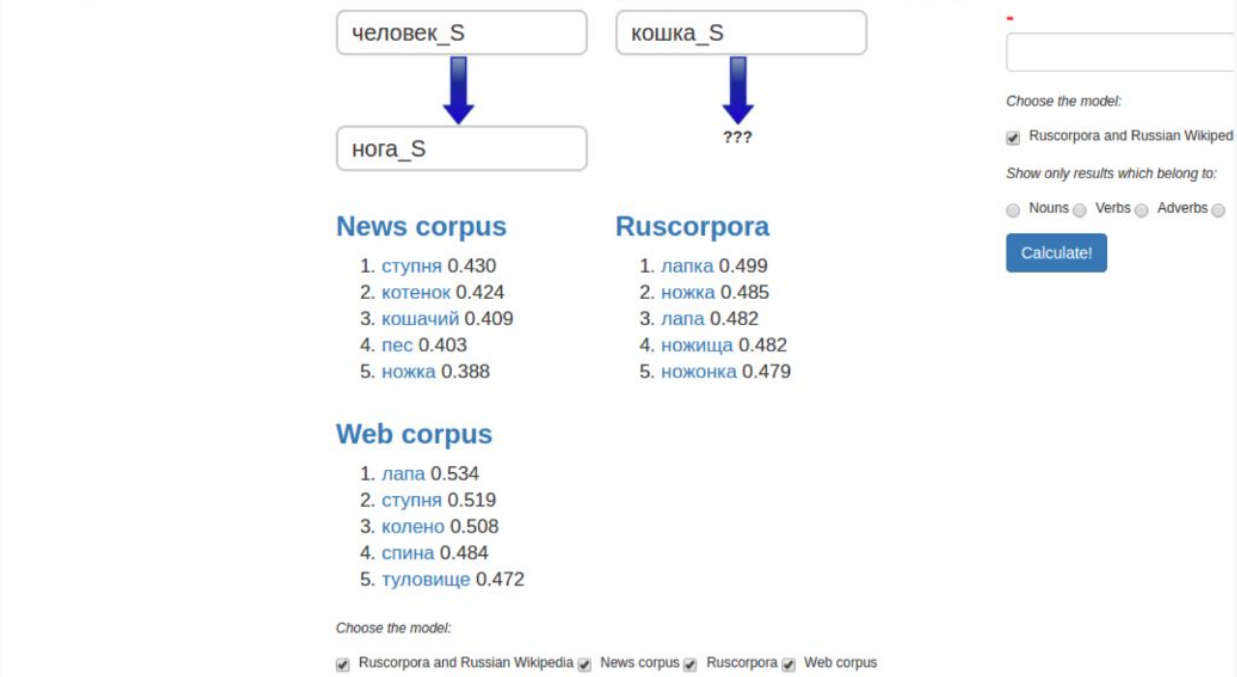
Проблемы word2vec

Невозможно установить тип семантических отношений между словами: синонимы, антонимы и т.д. будут одинаково близки, потому что обычно употребляются в схожих контекстах (например, слова ***хороший*** и ***плохой***).

Поэтому близкие в векторном пространстве слова называют семантическими **ассоциатами**. Это значит, что они семантически связаны, но как именно — непонятно.

Rusvectores, word2vec для русского

На rusvectores можно найти слова, наиболее близкие к данному, построить семантическую пропорцию и многое другое.



Где взять готовые эмбеддинги

Я рассказала, как обучить свои эмбеддинги. Но это долго, заморочно и не всегда нужно. Есть ли уже обученные эмбеддинги? Конечно!

Rusvectores! (для русских слов)

Ресурсы

Почитать

- про word2vec по-русски
- Introduction to Word Embedding and Word2Vec
- Word2Vec and FastText Word Embedding with Gensim
- про BPE
- word2vec tutorial на kaggle