МИСИС
УНИВЕРСИТЕТ

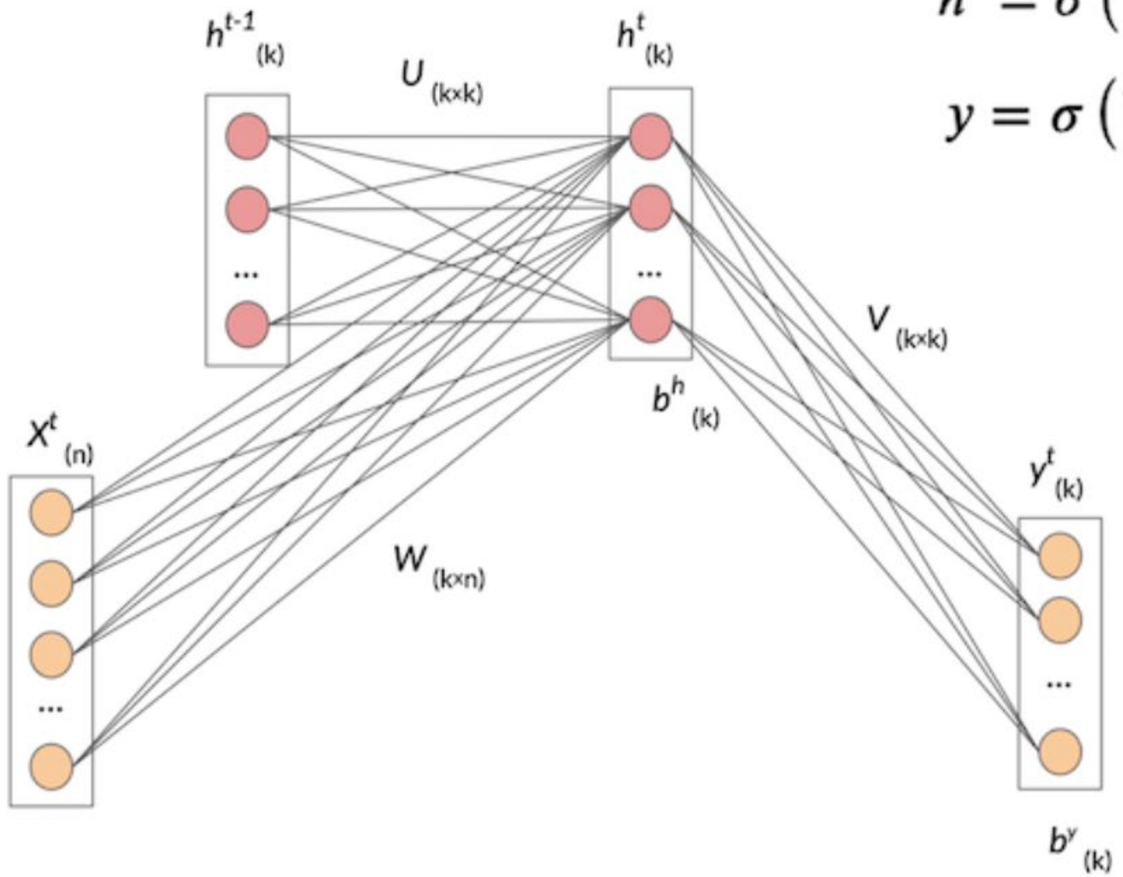
Архитектура seq2seq. Обработка текста

Корнет Мария Евгеньевна
старший преподаватель кафедры
Инженерная кибернетика

План

- Архитектуры кодировщик -декодировщик, seq2seq
- Текст в машинном обучении
- Методы представления текста
- Предобученные эмбеддинги, word2vec и др
- Контекстуальные представления CoV

Временной шаг работы RNN



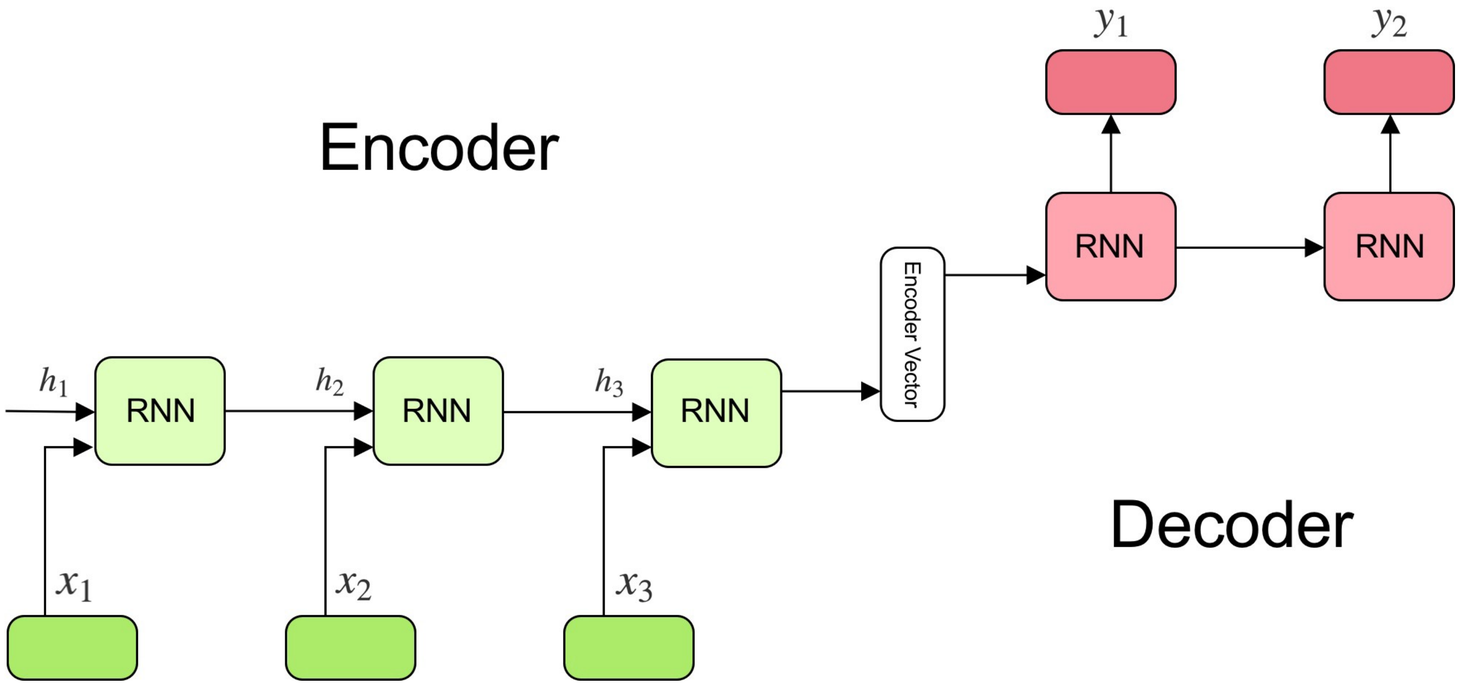
$$h^t = \sigma\left(WX^t + Uh^{t-1} + b_h\right)$$

$$y = \sigma\left(Vh^t + b_y\right)$$

- x^t — входной вектор в текущий момент времени;
- h^{t-1} — скрытое состояние из прошлого шага;
- h^t — новое скрытое состояние;
- y^t — выходной вектор (например, предсказание).

Архитектура Seq2Seq

Seq2Seq (**Sequence to Sequence**) — это тип архитектуры нейронной сети, для преобразования одной последовательности данных в другую.



https://www.davidsbatista.net/blog/2020/01/25/Attention-seq2seq/

Классический Seq2Seq

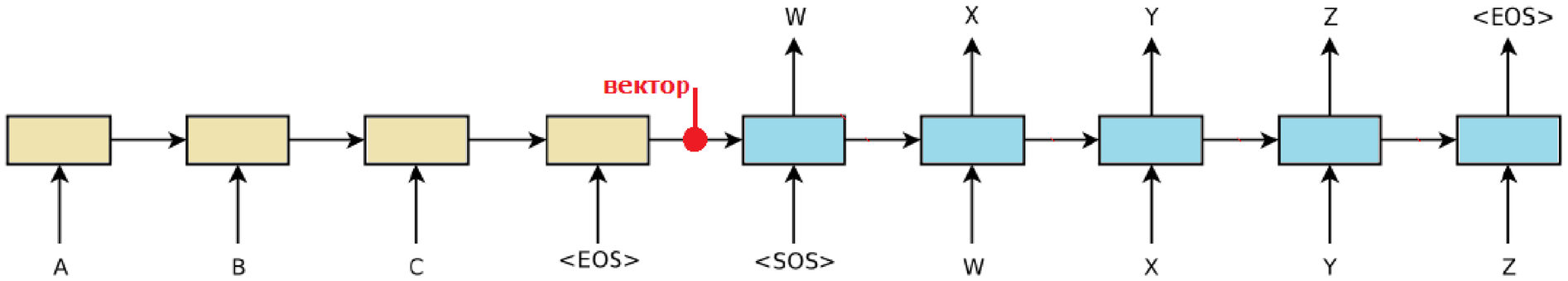
Энкодер читает входную последовательность (например, A, B, C) и преобразует её в **один вектор фиксированной длины** — так называемый **контекстный вектор.**

Декодер использует этот вектор, чтобы сгенерировать выходную последовательность (W, X, Y, Z...).

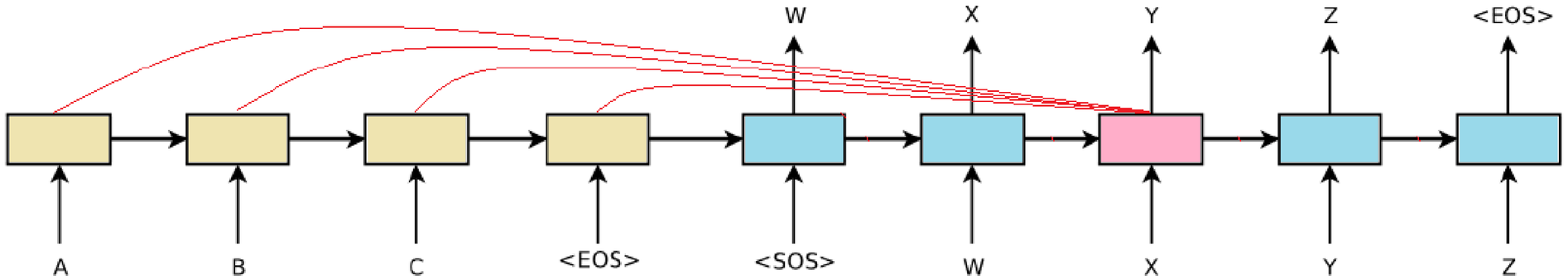
Причем RNN-блоки энкодера и декодера имеют разные параметры!

Проблема: **Вся информация о входной последовательности хранится в одном векторе!**

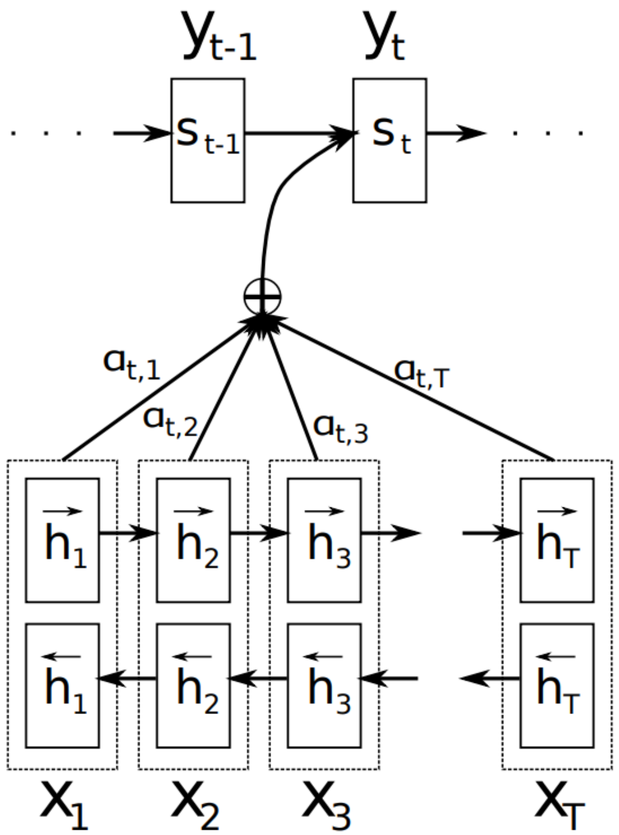
Механизм внимания



Решение – механизм внимания



Механизм внимания seq2seq



Энкодер обрабатывает вход $X_1, X_2, .., X_T$, считает скрытые состояния h_i.

Декодер на каждом шаге t вычисляет веса внимания:

1. $$e_{t,i} = \text{score}(s_{t-1}, h_i) \quad \Rightarrow \quad a_{t,i} = \frac{\exp(e_{t,i})}{\sum_j \exp(e_{t,j})}$$

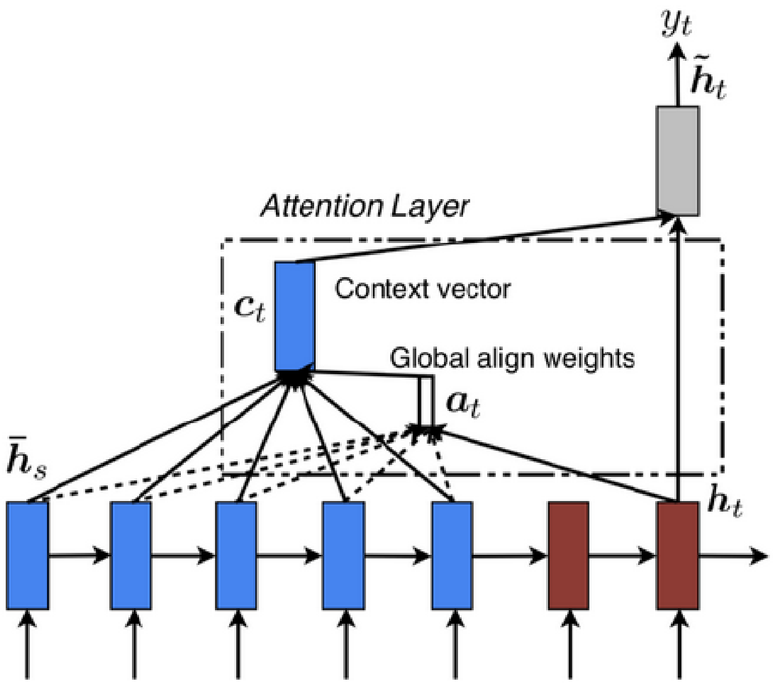
   Softmax по всем позициям входа.

2. Вычисляет контекстный вектор c_t:

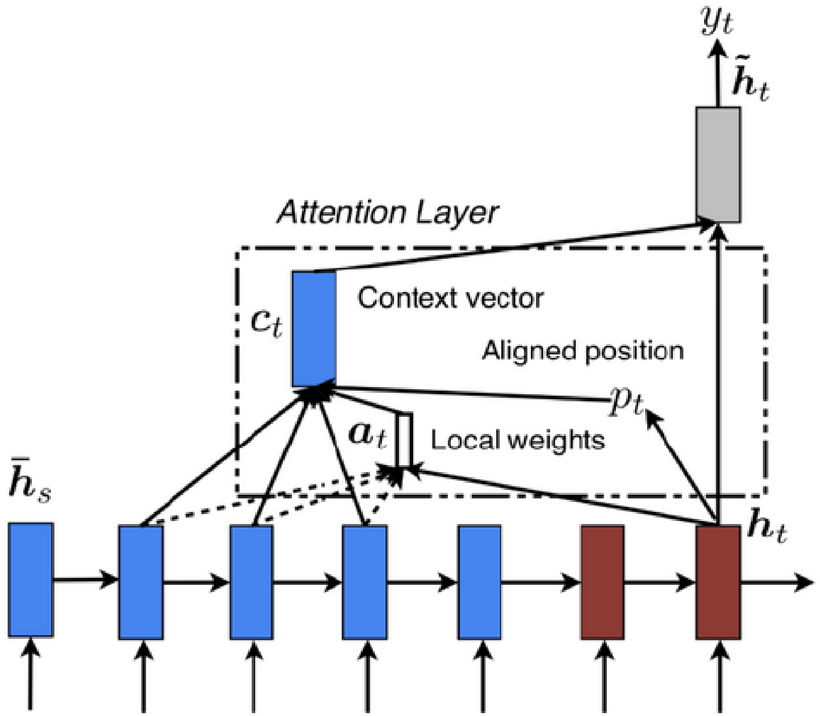
$$c_t = \sum_{i=1}^{T} a_{t,i} \cdot h_i$$

3. Контекст c_t вместе с предыдущим состоянием s_{t-1} подаётся на декодер для вычисления нового состояния s_t

Виды внимания



Global Attention Model



Local Attention Model

Декодер **смотрит на все скрытые состояния энкодера**

Декодер **смотрит только на ограниченное окно** из входных скрытых состояний

Особенности текста

Текст представляет собой последовательность токенов из конечного алфавита.

Токен в контексте обработки естественного языка — элементы текста (слова, части слов, знаки препинания), на которые разбивается текст при обработке.

Словарь — набор всех допустимых токенов, с которыми работает модель. Он может быть фиксированным или адаптивным.

Особенности:

- Текст может быть разной длины.
- В тексте присутствует ***временная компонента***: текст читается слева направо (или в другом направлении в зависимости от языка).

Особенности текста

- **Текст — дискретный сигнал**, насыщенный многозначностью.

Слова могут иметь несколько значений (полисемия) или звучать одинаково, но обозначать разные вещи (омонимы), что создаёт сложности при интерпретации.

- Язык постоянно эволюционирует. Появляются **неологизмы, заимствования**. Понимание таких слов требует культурного и контекстуального знания.

- **Устойчивые выражения** и **профессиональный слэнг**.

- Один и тот же смысл передаётся по разному.

Обработка естественного языка

NLP (Natural Language Processing) - это область, изучающая автоматическую обработку и анализ естественного языка. Сюда входят задачи от простого разбиения текста на слова до глубокого понимания смысла.

Обработка текста фокусируется на анализе структуры (разбиение на слова, подсчёт частот, перевод в числа) и содержания текста, а **понимание** — на интерпретации смысла.

Понимание текста — это способность модели:

- ✓ определить, **о чём говорится** в тексте;
- ✓ выделить **ключевые объекты** и **события**;
- ✓ выявить **связи между предложениями**;
- ✓ сделать **уместные выводы**;
- ✓ ответить на **вопросы по тексту**;
- ✓ распознать **тональность, иронию, эмоции**.

Задачи с текстами

Анализ тональности: определение эмоциональной окраски (положительная, нейтральная, отрицательная). Пример: отзывы о фильмах.

Определение авторства: установление, кто написал текст по стилю или лексике.

Машинный перевод: автоматический перевод между языками (например, DeepL, Google Translate).

Аннотирование: краткое изложение длинных текстов.

Вопросно-ответные системы: как в чат-ботах или голосовых помощниках.

OCR (оптическое распознавание символов): преобразование изображений с текстом в машинно-читаемый формат.

Извлечение объектов: определение именованных сущностей — людей, мест, дат и т. д.

Датасеты

Текстовые данные требуют **ручной разметки.**

Вопросно-ответные системы — одна из самых востребованных задач. Пример: найти в тексте, из чего состоит атом.

SQuAD (Stanford Question Answering Dataset) — один из самых известных и широко используемых датасетов.
SberQUAD — русскоязычный датасет для вопросно-ответных систем, созданный Сбером по аналогии со Stanford QA Dataset (SQuAD).

WikiQA — открытый датасет для ответов на вопросы на основе поисковых запросов Bing и страниц Wikipedia.

WikiMovies — вопросно-ответный датасет в домене кино, связанный с базами знаний из Wikipedia.

WebQuestions — датасет вопросов, сгенерированных на основе Google Suggest API, с ответами из Freebase.

Датасеты

Анализ тональности:

Amazon Review Data — набор отзывов о продуктах с метками, указывающими на эмоциональную окраску: положительную, отрицательную или нейтральную.

Перевод:

Bible Corpus — параллельный корпус, созданный из переводов Библии, включает 102 языка.

Краткое изложение:

Wikipedia Current Events Portal (WCEP) Dataset — набор коротких резюме о новостях, написанных человеком, каждый из которых связан с кластером новостных статей, связанных с событием.

Датасеты

Классификация:

Spam SMS Collection — коллекция SMS-сообщений, размеченных как спам или не спам (ham). Используется для задач бинарной классификации — спам/не спам.

Fake News Classification — датасет для классификации фейковых новостей: содержит около 72 000 новостных статей, из которых 35 000 — реальные и 37 000 — фейки.

AG's News Topic Classification: Популярный датасет для классификации новостных статей на 4 темы: "World", "Sports", "Business", "Sci/Tech".

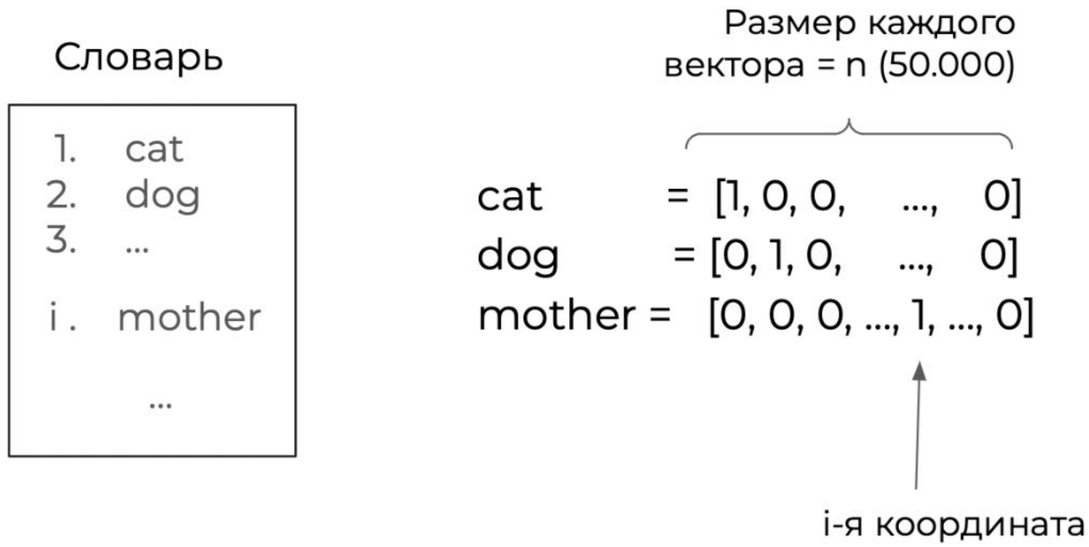
Reuters-21578 (R8 & R52): Еще один классический датасет новостных статей от Reuters. Часто используются его подмножества R8 (8 тем) и R52 (52 темы).

RuSentiment: Русскоязычный датасет для анализа тональности, содержащий твиты с размеченными эмоциями (позитивный, негативный, нейтральный) и более детальными категориями.

Представление текста

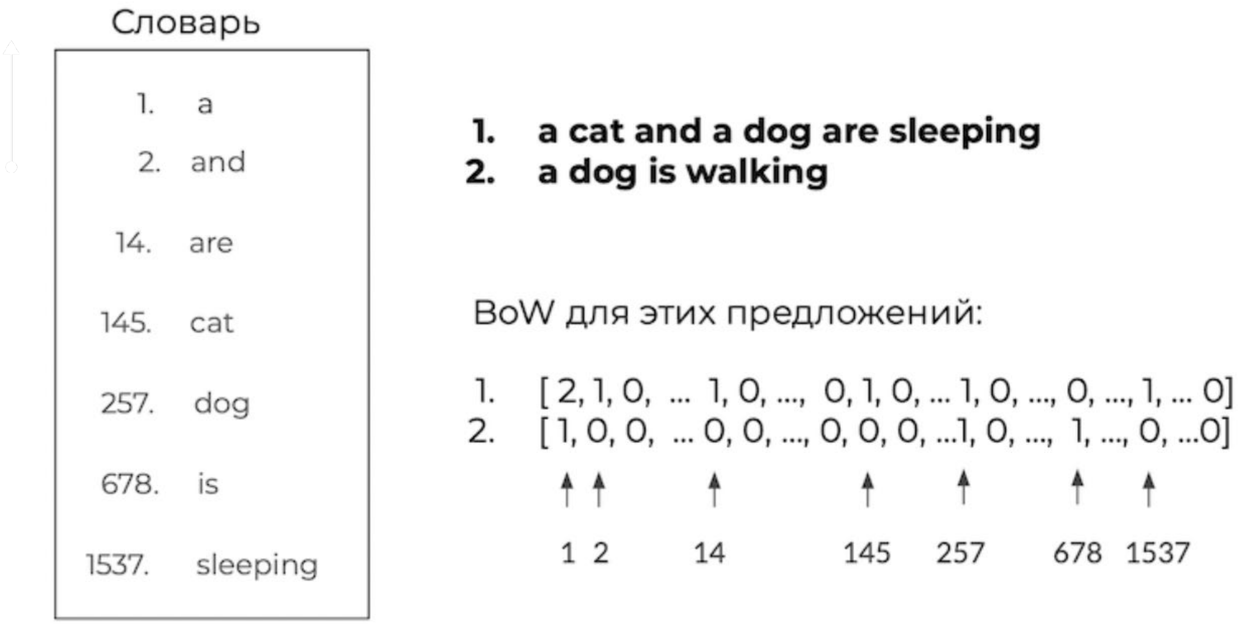
Векторные представления слов (эмбеддинги, Word Embeddings) — это числовые векторы, которые кодируют семантическое и синтаксическое значение слов. Слова с похожим значением имеют близкие векторные представления в многомерном пространстве.

OHE (One-Hot Encoding):



Мешок слов

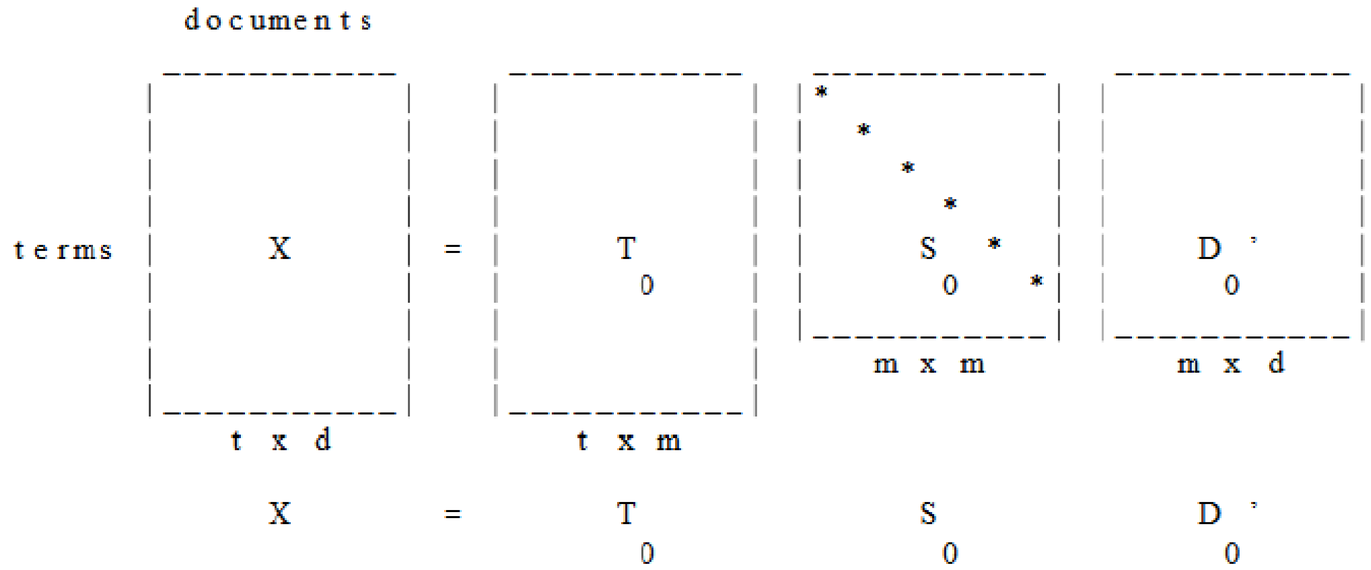
Bag-of-Words:



1. **Составление словаря** — определяется список всех уникальных слов во всём наборе документов.
2. **Векторизация** — каждый отдельный текст представляется как вектор, где указывается количество вхождений каждого слова из словаря.

Латентный семантический анализ

LSA (Latent Semantic Analysis):



1.Создается матрица "термины-документы", где строки — слова, столбцы — документы, значения — частоты слов.

2.Применяется **SVD (сингулярное разложение)** для уменьшения размерности матрицы.

3.Убираются "шумовые" компоненты, остаются только семантически значимые факторы.

http://wordvec.colorado.edu/

TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF)

TF (Частота термина) обозначает, насколько часто определенное слово появляется в данном документе. Таким образом, TF измеряет важность слова в контексте отдельного документа.

IDF (Обратная частота документа) измеряет, насколько уникально слово является по всей коллекции документов. Слова, которые появляются в большинстве документов, имеют низкое IDF, так как они не вносят большой информационной ценности.

TF-IDF(t, d) = TF(t, d) * IDF(t)
где:
TF(t, d) - Частота термина (TF) для слова "t" в документе "d".
IDF(t) - Обратная частота документа (IDF) для слова "t".

https://habr.com/ru/companies/otus/articles/755772/

Эмбенддинги

Эмбеддинги (embeddings) основаны на идее, что **смысл слова определяется его окружением**, то есть контекстом.

Word2Vec — это метод обучения векторных представлений слов, разработанный в Google в 2013 году.

Принцип **дистрибутивной семантики** (distributional semantics):

«Скажи мне, с кем ты употребляешься, и я скажу, кто ты».

1. Маша ездит на _____
2. Колесо _____ было проколото
3. У _____ красивая белая рама

	1	2	3
Велосипед	+	+	+
Мотоцикл	+	+	+
Машина	+	+	-
Лошадь	+	-	-

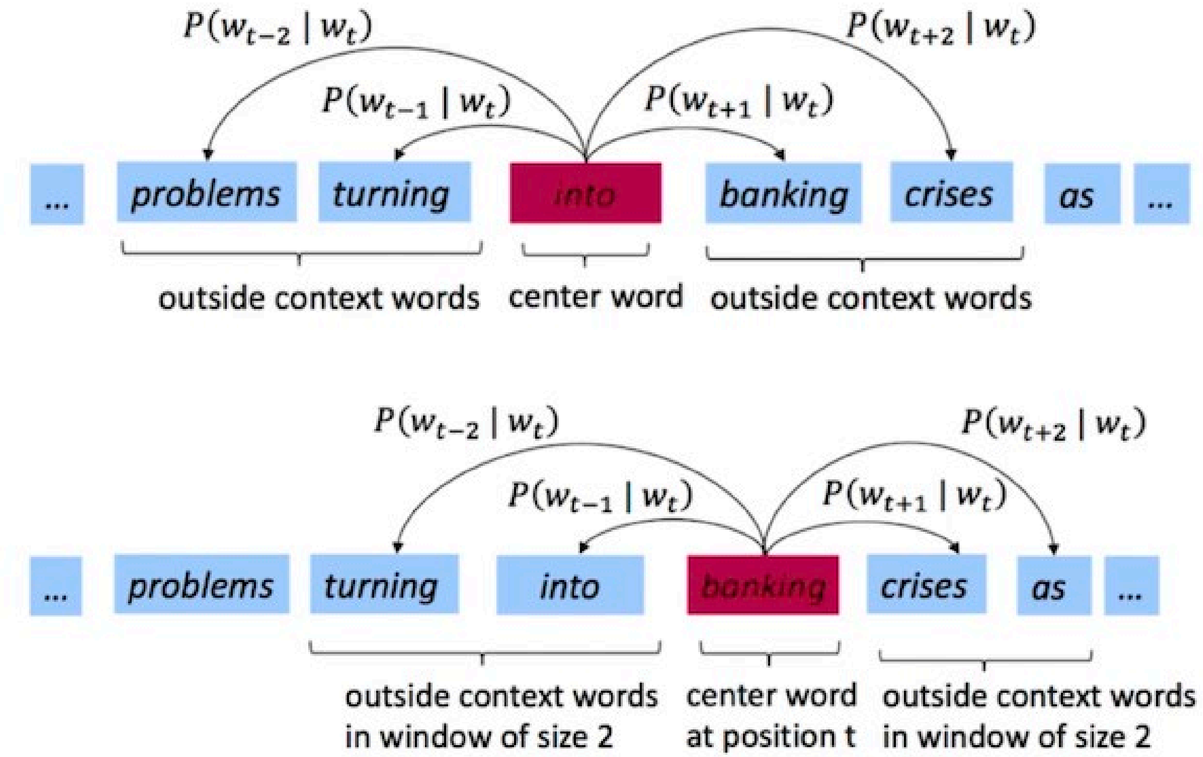
Матрица совместной встречаемости

	a	horse	ride	bicycle	frame	rose
a		256	45	230	90	134
horse	256		137	4	2	5
ride	45	137		120	34	3
bicycle	230	4	120		76	2
frame	90	2	34	76		0
rose	134	5	3	2	0	

Строки и столбцы — уникальные слова из словаря размером n.
Элемент M[i][j] — количество раз, когда слово j встречалось в **контекстном окне** слова i.
Контекст – слово, которое недалеко располагается (в окрестности).

Проблемы такого подхода:
1. **Высокая размерность** — n×n при большом словаре.
2. **Разреженность** — большинство ячеек нулевые.
3. **Не учитывает семантику** — только статистическую близость.

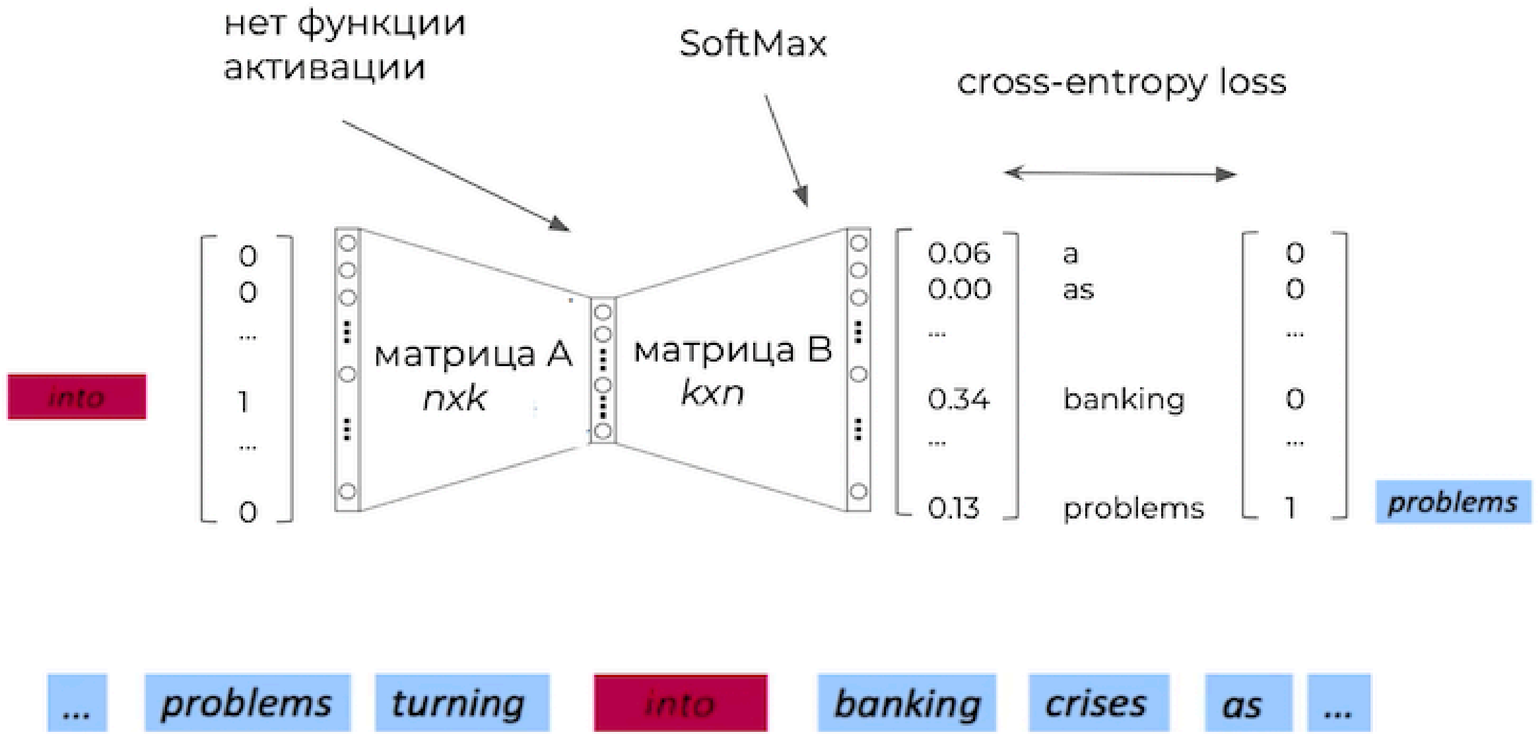
Word2vec



Идея: обучить учить нейросеть по слову предсказывать слова, которые могут находиться в контексте.

Ставится задача классификации – количество классов – *размер словаря.*

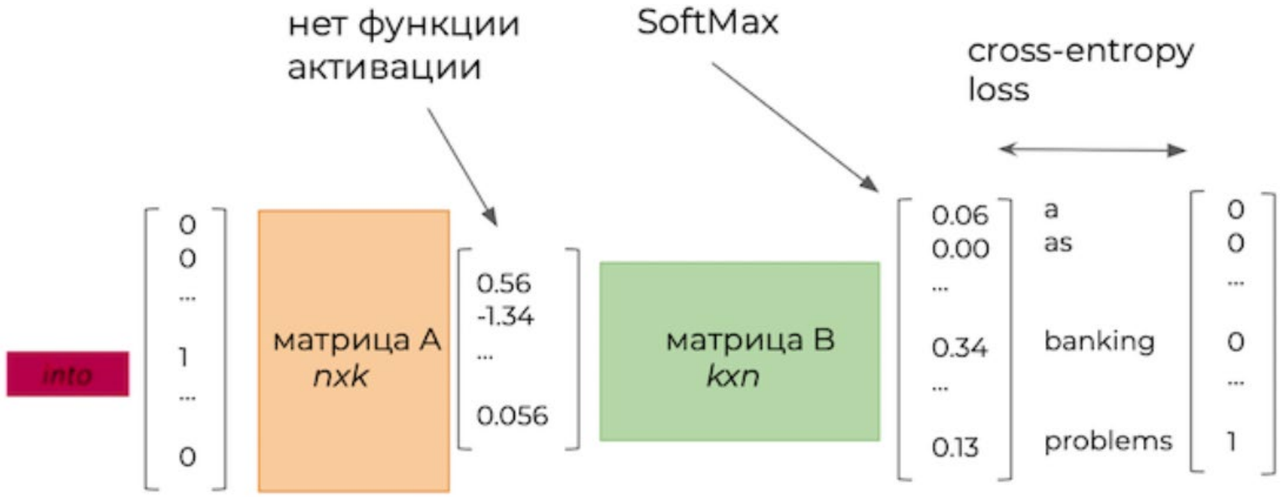
Word2vec

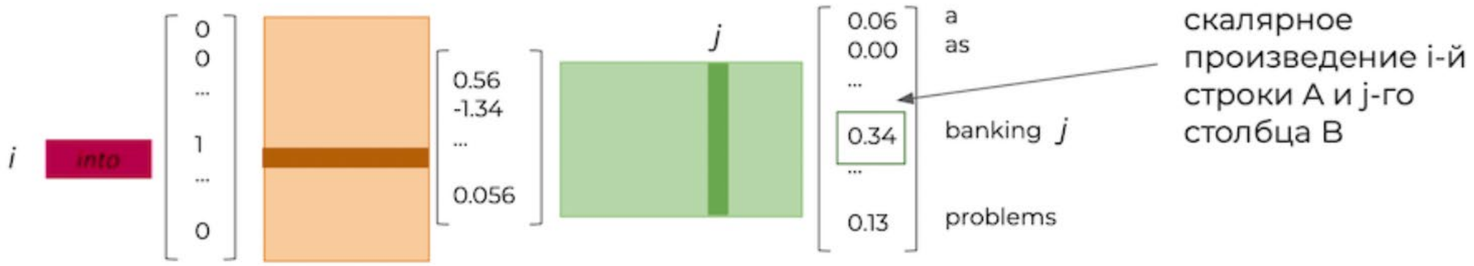


- На вход нейросеть принимает слово, его one-hot вектор.
- На выходе выдает n значений — распределение на слова в словаре.
- Loss — кросс-энтропия между распределением, выданным сетью, и верным распределением (one-hot вектором).

Эмбеддингов слов можно взять строки матрицы A или столбцы матрицы B.

Word2vec





Эмбеддинги:

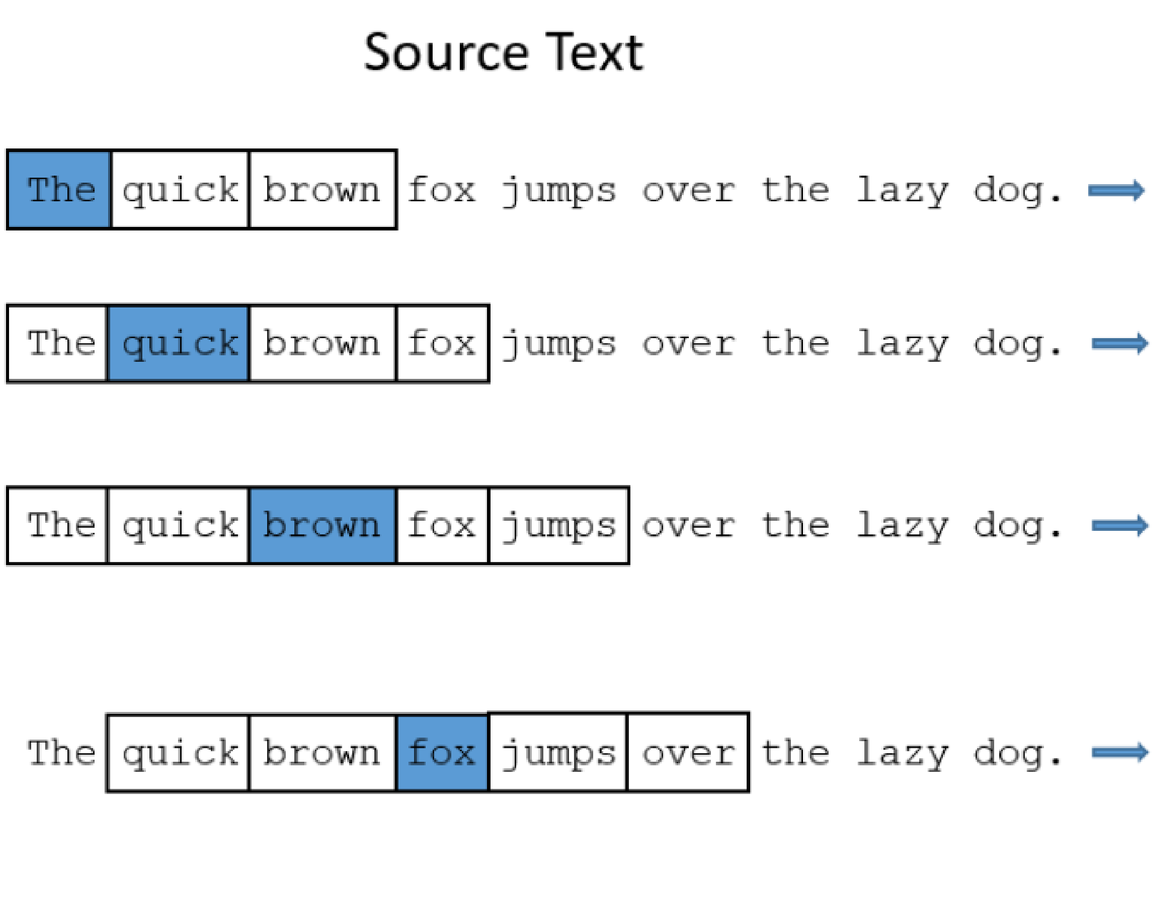
- Строки матрицы A,
- Столбцы матрицы B

Сравнивать эмбеддинги можно по скалярному произведению, например косинусное сходство:

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

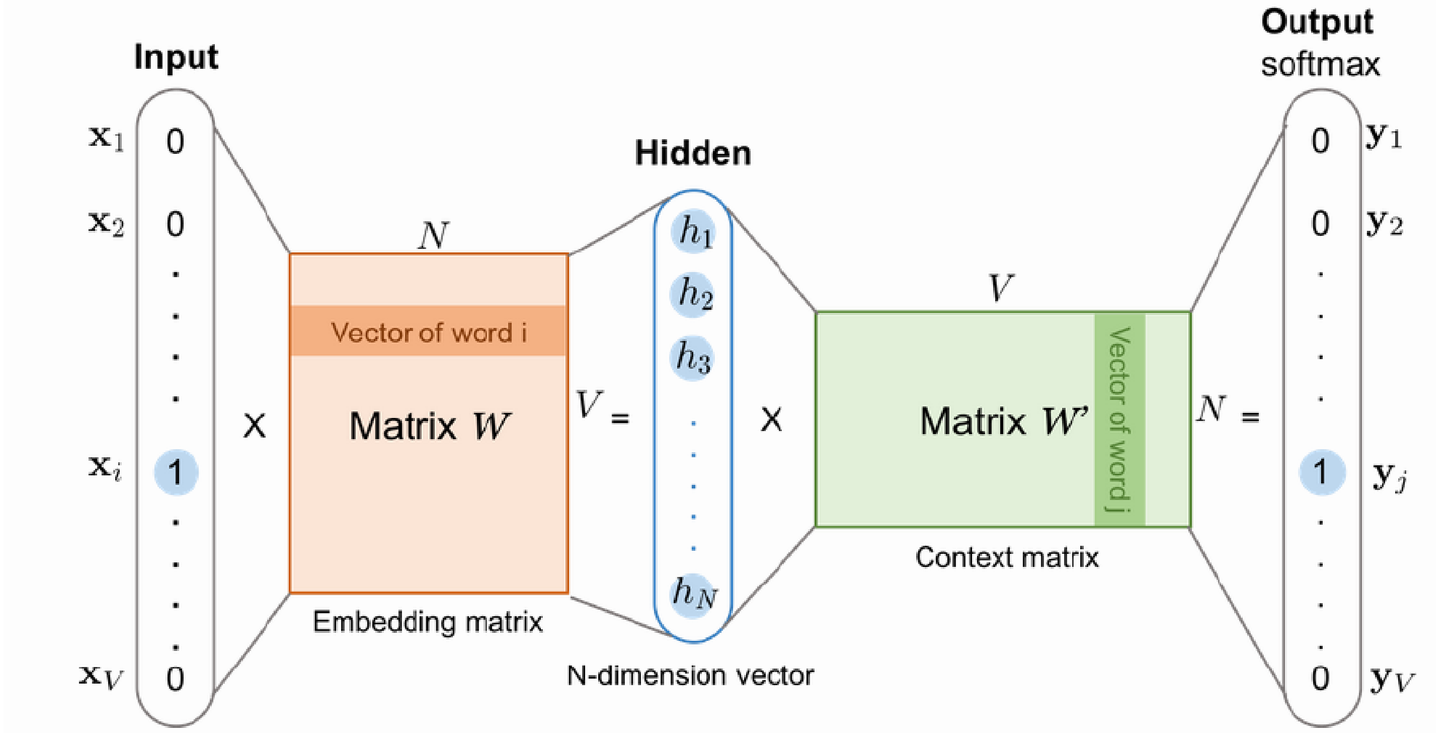
Skip-gram

Предсказание контекста по центральному слову



Центральное слово → контекстные слова

Skip-gram

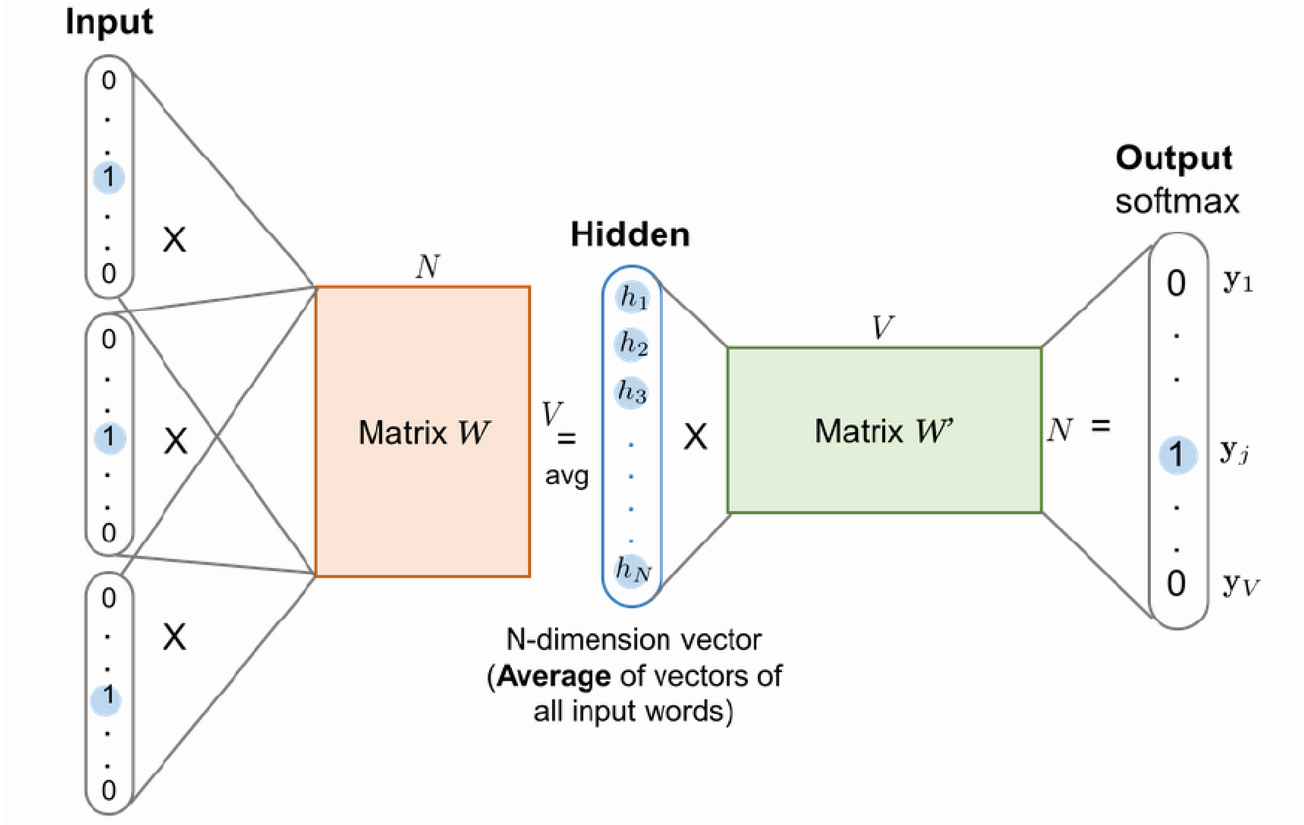


Вход – ОНЕ кодировка,
Выход – распределение вероятностей.

https://lilianweng.github.io/posts/2017-10-15-word-embedding/

CBOW

Continuous Bag Of Words: Предсказание центрального слова по словам контекста.



Контекстные слова → центральное слово

https://lilianweng.github.io/posts/2017-10-15-word-embedding/

Техника негативного сэмплирования

Цель:
Обучить модель отличать «настоящие» пары (центральное слово и слово из контекста) от **случайных (негативных) пар**, которые не встречаются рядом в реальном тексте.

Логарифмическая функция потерь:

$$\log(1 + \exp(-s(x_t, x_c))) + \sum_{x \in N_{t,c}} \log(1 + \exp(s(x_t, x)))$$

Здесь:

- x_t — центральное слово (target).
- x_c — правильное контекстное слово.
- $N_{t,c}$ — множество **негативных примеров** (слова, которые не должны быть в контексте).
- $s(x_t, x_c)$ — **скоринг-функция**, вычисляющая близость пары слов (обычно скалярное произведение их векторов).

Другие методы

FastText Идея в том, чтобы строить эмбеддинги не для целых слов, а для частей.
Конструирует эмбеддинг всего слова из эмбеддингов его частей.
Можно получать эмбеддинги многих слов, которых нет в словаре.

GloVe (Global Vectors) использует статистическую информацию о частоте встречаемости слов и фраз в тексте, чтобы улучшить обучение эмбеддингов редких слов.

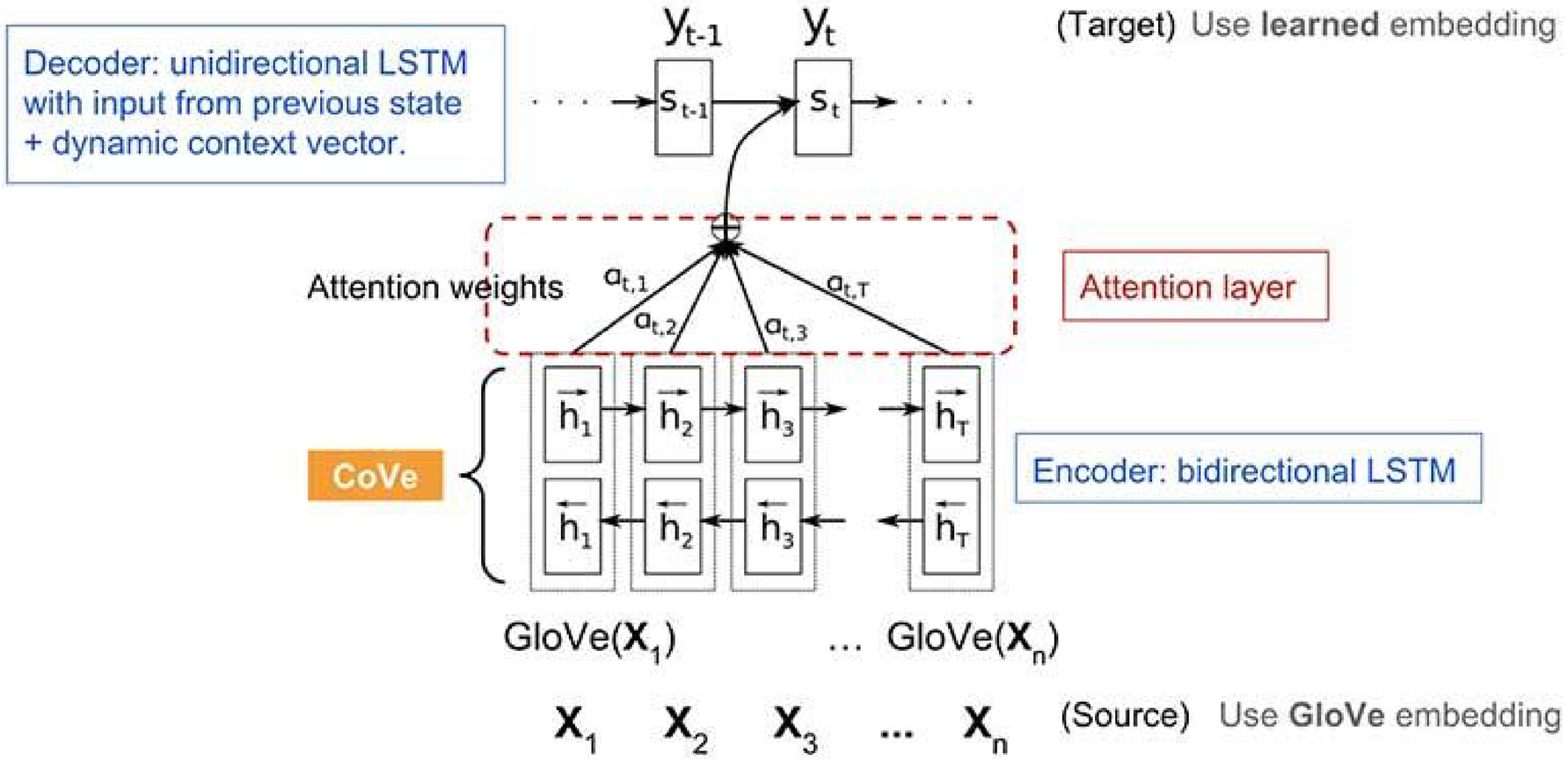
CoVe

CoVe (Contextualized Word Vectors) — метод контекстуальных представлений слов, разработанный компанией SalesForce Research в 2017 году.
Основная идея CoVe заключается в переносе знаний.

Тип представления текста	Основная характеристика	Пример
Статические векторы(Word2Vec, GloVe)	Каждому слову соответствует один фиксированный вектор, независимо от контекста.	Слово **"ключ"** будет иметь одно и то же представление в контекстах:– "ключ от двери"– "скрипичный ключ".
Контекстуальные векторы(CoVe и др.)	Представление слова динамически формируется в зависимости от окружающих его слов в предложении.	Для слова **"ключ"** в контекстах:– "ключ от двери"– "скрипичный ключ" будут сгенерированы **разные векторы**.

CoVe

Contextual Word Vector: Кодирование слова зависит от контекста

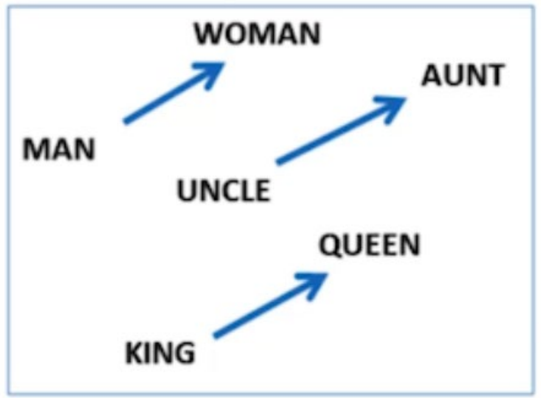


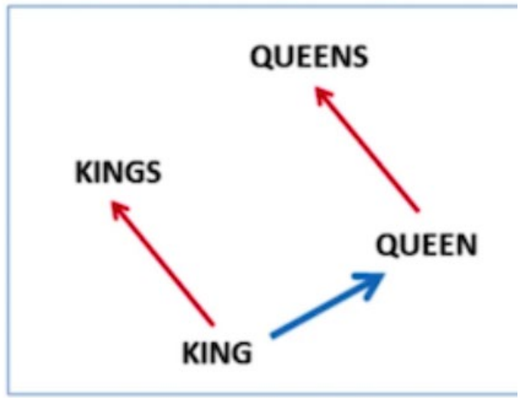
Проверка представлений

Таблицы соответствия (аналогия слов):

king – man + woman = queen

1. Расчёт вектора: vec("king") – vec("man") + vec("woman")
2. Сравнение полученного вектора с другими словами по косинусному сходству.

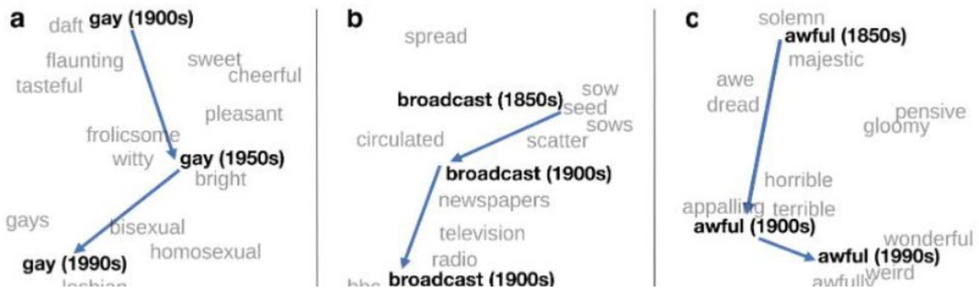




Проверка представлений

Семантический сдвиг:

Обучаем модель на текстах разных временных периодов.
2. Сравнение полученного вектора с другими словами по косинусному сходству.



МИСИС
УНИВЕРСИТЕТ

Спасибо
за внимание!