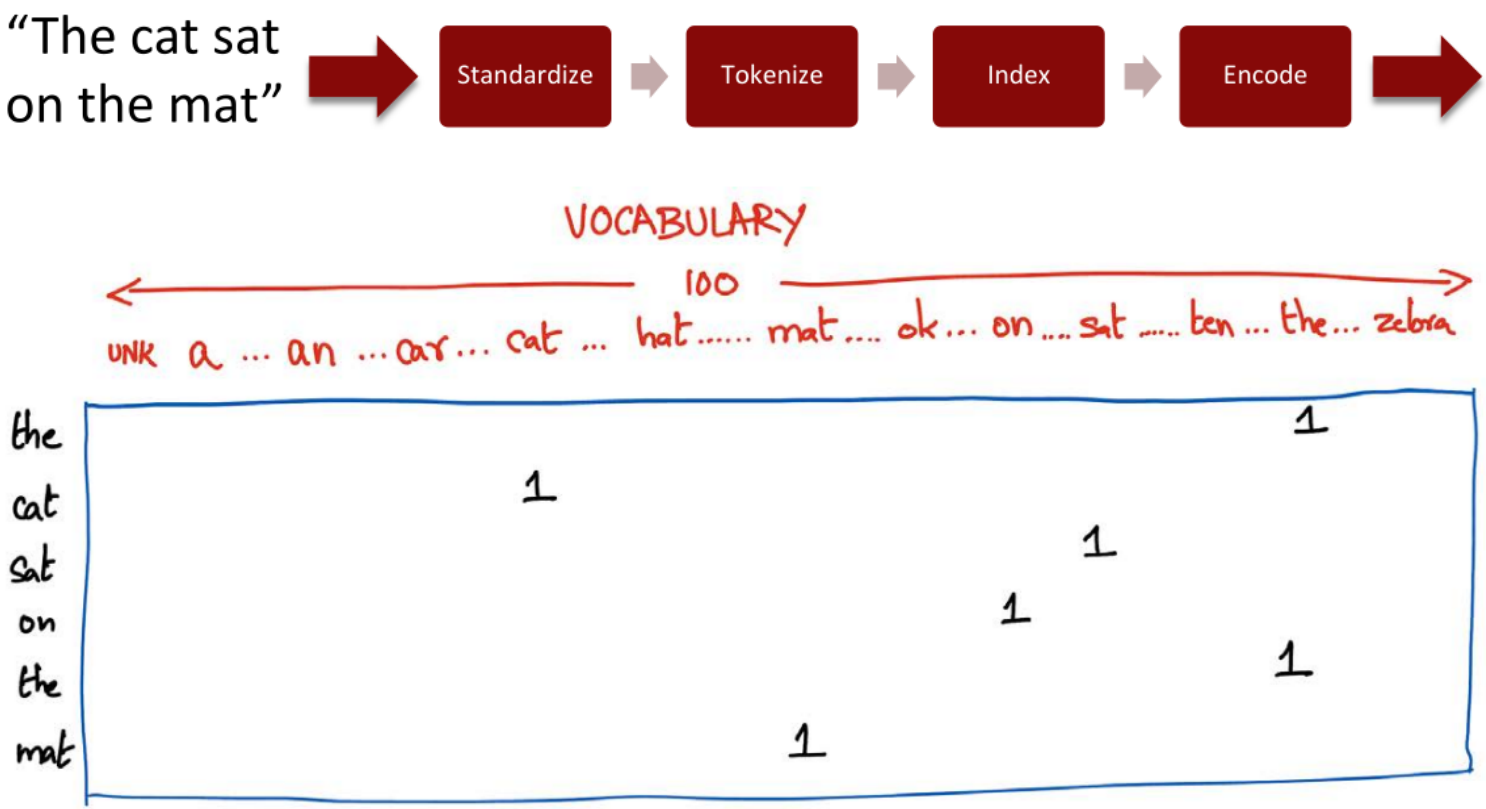
Лекция 6: Встраивание

26 апреля 2026 г.

До настоящего времени мы кодировали входные текстовые строки с помощью one-hot векторов.

До настоящего времени мы кодировали входные текстовые строки с помощью one-hot векторов.



Проблема с векторами one-hot

Проблема с векторами one-hot (1/2)

Если словарь очень длинный, каждый токен будет иметь one-hot вектор, длина которого равна размеру словаря.

- Эту проблему можно частично смягчить, выбрав только наиболее часто встречающиеся слова.
- Тем не менее, это увеличивает количество весов, которые модель должна изучить, и, следовательно, увеличивает время вычислений, а также риск переобучения.

Проблема с векторами one-hot(2/2)

- Предположим, мы создали словарь на основе обучающего корпуса.

Проблема с векторами one-hot(2/2)

- Предположим, мы создали словарь на основе обучающего корпуса.
- Рассмотрим векторы, закодированные в формате one-hot для слов «movie» и «film».
  - Находятся ли эти два вектора «близко» друг к другу?

Проблема с векторами one-hot(2/2)

- Предположим, мы создали словарь на основе обучающего корпуса.
- Рассмотрим векторы, закодированные в формате one-hot для слов «movie» и «film».
  - Находятся ли эти два вектора «близко» друг к другу?
- А что насчет векторов, закодированных в формате one-hot для слов «good» и «bad»?
  - Они находятся «далеко» друг от друга?

Проблема с векторами one-hot(2/2)

- Предположим, мы создали словарь на основе обучающего корпуса.
- Рассмотрим векторы, закодированные в формате one-hot для слов «movie» и «film».
  - Находятся ли эти два вектора «близко» друг к другу?
- А что насчет векторов, закодированных в формате one-hot для слов «good» и «bad»?
  - Они находятся «далеко» друг от друга?
- Расстояние между любыми двумя векторами, закодированными в формате one-hot, одинаково независимо от слов! Это никак не связано со «значением» слов.

Резюме: Проблемы с one-hot векторами

- Если словарь очень длинный, каждый токен будет иметь вектор one-hot, длина которого равна размеру словаря.
- Между значением слова и его вектором one-hot отсутствует какая-либо связь.

Было бы здорово, если бы...

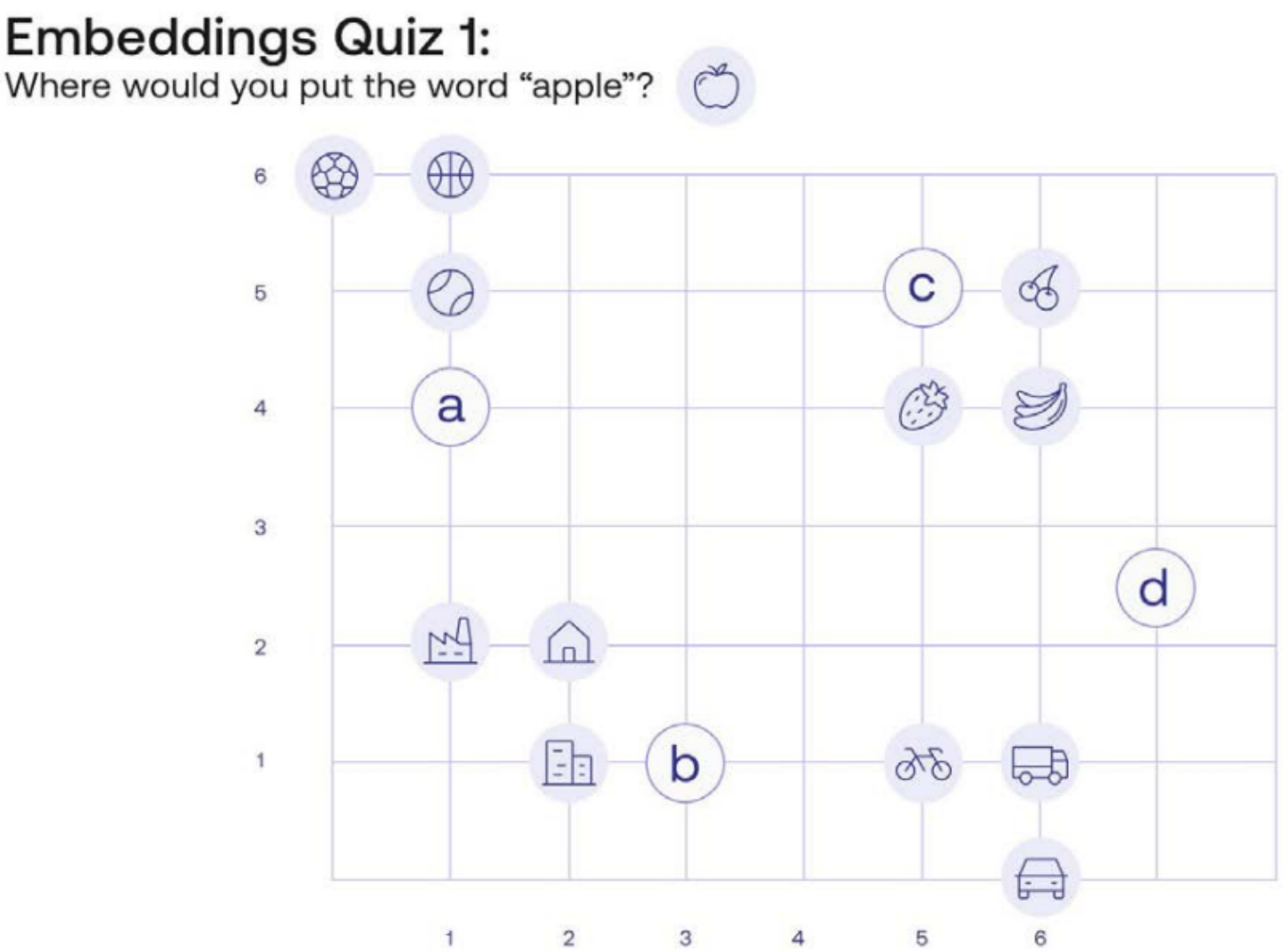
Было бы здорово, если бы. . .

- Векторы, представляющие синонимы (например, movie и film) или связанные слова (например, apple, banana), расположены близко друг к другу.

Было бы здорово, если бы. . .

- Векторы, представляющие синонимы (например, movie и film) или связанные слова (например, apple, banana), расположены близко друг к другу.
- Векторы для слов, имеющих совершенно разное значение, находятся очень далеко друг от друга.

Где вы разместите слово «apple»: в точке a, b, c или d?



Где вы разместите слово «apple»: в точке a, b, c или d?

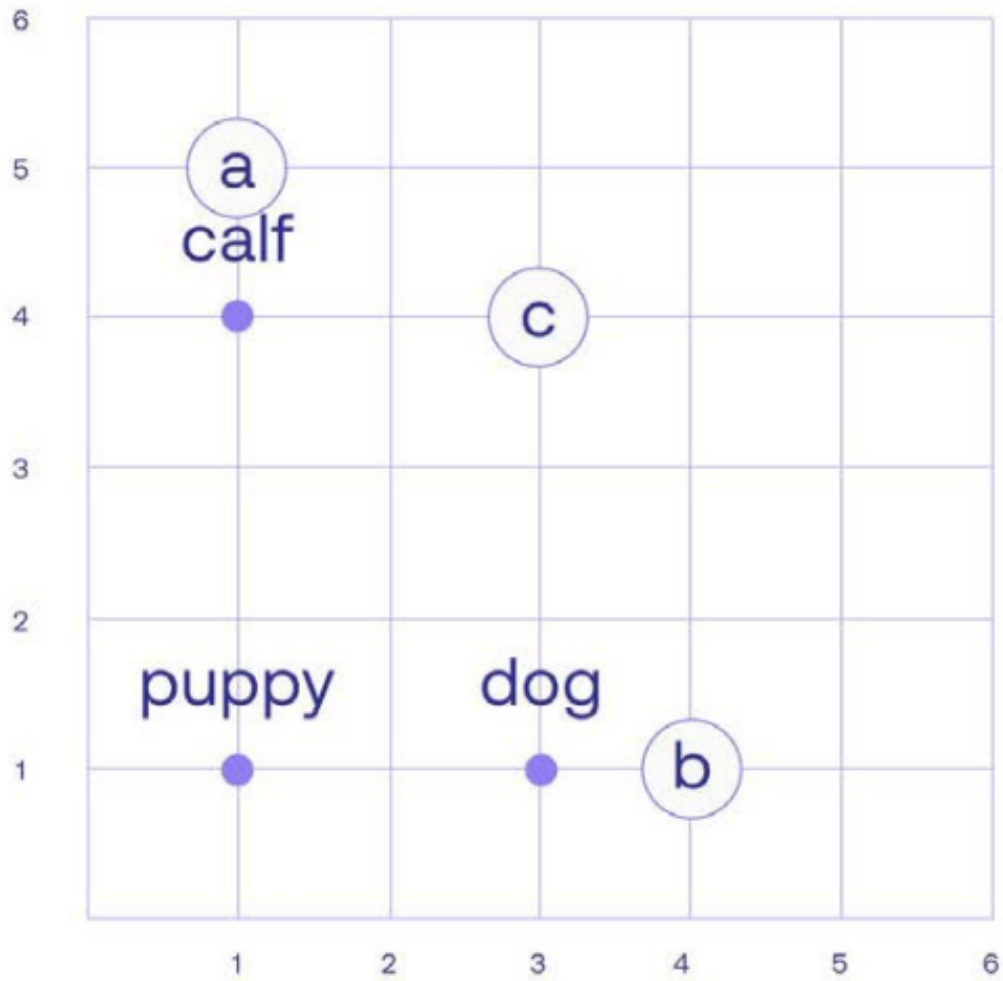
Было бы здорово, если бы...

- Векторы, представляющие синонимы (например, movie и film) или связанные слова (например, apple, banana), расположены близко друг к другу.
- Векторы для слов, имеющих совершенно разное значение, находятся очень далеко друг от друга.
- В более общем смысле: разве не было бы замечательно, если бы геометрическая связь между векторными представлениями слов отражала «семантическую связь» между словами?

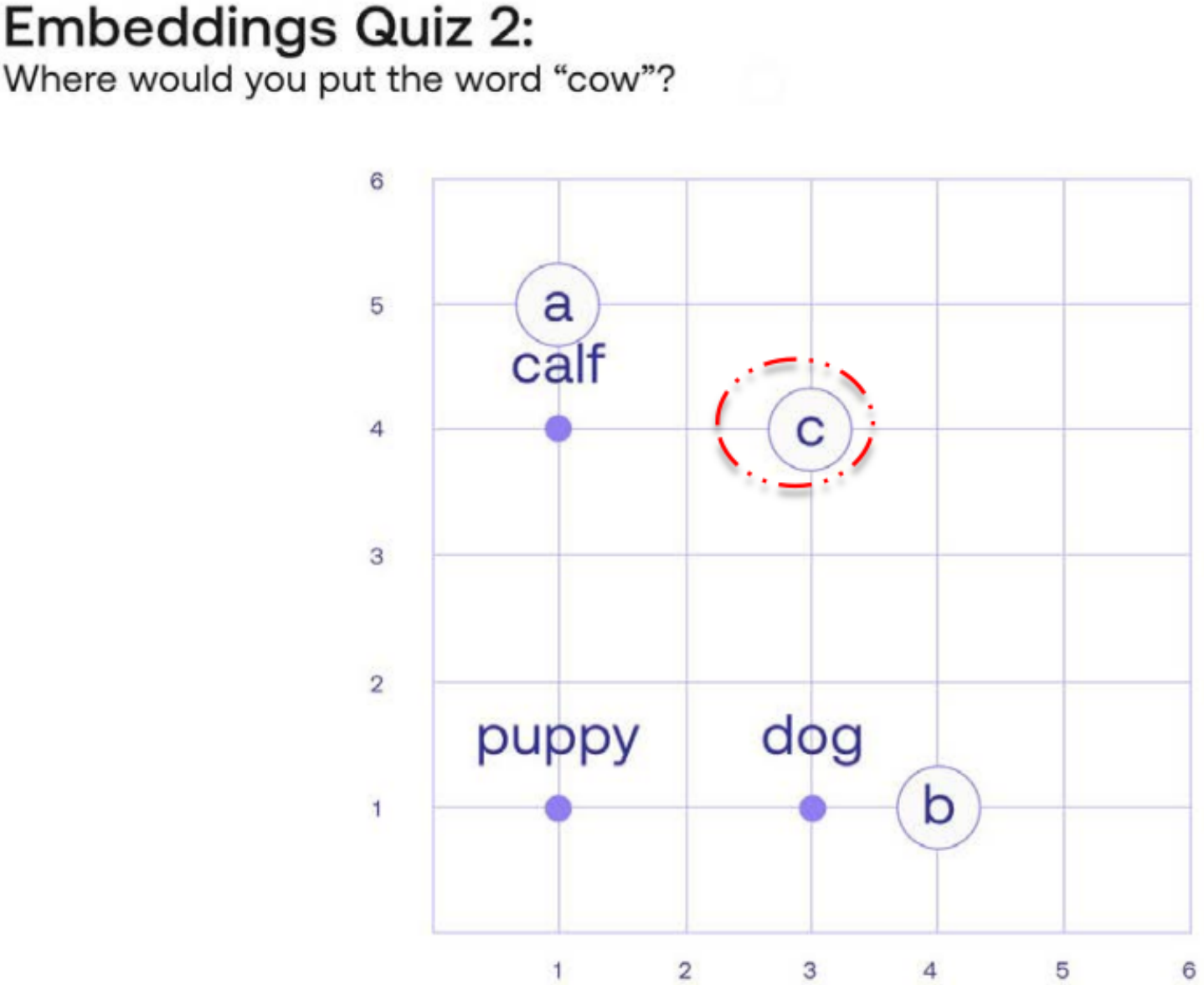
Куда вы поместите слово «cow»?

Embeddings Quiz 2:

Where would you put the word “cow”?



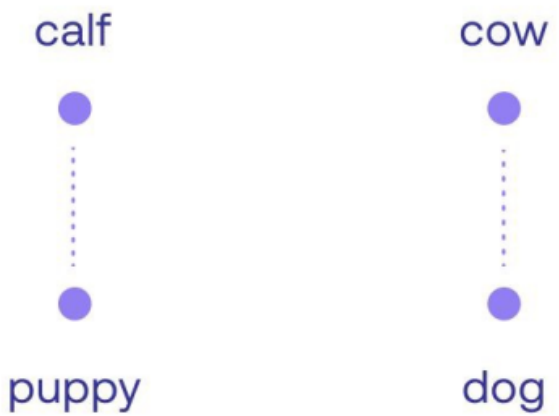
Куда вы поместите слово «cow»?



Семантическое расстояние

A puppy is to a dog, like a calf is to a cow

A puppy is to calf, like a dog is to a cow



Вложение слов — это векторы слов, предназначенные именно для достижения этой цели...

- Векторы, представляющие синонимы (например, movie и film) или связанные слова (например, apple, banana), расположены близко друг к другу.
- Векторы для слов, имеющих совершенно разное значение, находятся очень далеко друг от друга.
- В более общем смысле: разве не было бы замечательно, если бы геометрическая связь между векторными представлениями слов отражала «семантическую связь» между словами?

...и решить обе эти проблемы

- Если словарь очень длинный, каждый токен будет иметь one-hot вектор, длина которого равна размеру словаря.
- Нет никакой связи между значением слова и его one-hot вектором.

- Но нам нужно сделать еще кое-что.
- Одно и то же слово может означать разные вещи в зависимости от окружающих его слов.
- Таким образом, нам нужно найти способ сделать вложение контекстуальное (то есть, учитывать другие слова в предложении).

Контекстные вложение слов — это вектор слова, которые удовлетворяют обоим этим «требованиям».

- Геометрическая взаимосвязь между векторами слов должна отражать «семантическую связь» между словами.
- Вектор слова должен учитывать другие слова в предложении, то есть он должен принимать во внимание контекст слова.

Ключ к вычислению вложнения контекстуального слов = трансформеры!

Сегодня мы рассмотрим, как вычислять вложения отдельных слов.

На следующей неделе: контекстные вложения слов с помощью трансформеров!

Как можно получить вложение слова из данных.

- Мы можем вручную собрать синонимы, антонимы, связанные слова и т. д. и попытаться присвоить им вложения слов, удовлетворяющие нашим требованиям.

Как можно получить вложение слова из данных.

- Мы можем вручную собрать синонимы, антонимы, связанные слова и т. д. и попытаться присвоить им вложения слов, удовлетворяющие нашим требованиям.
- Но есть ли лучший способ? Можно ли как-то узнать всё это из данных без ручных усилий?

Мы можем!

Главная мысль:

“Вы узнаёте слово по компании, которой оно придерживается”

Джон Ферт

“Вы узнаёте слово по компании, которой оно придерживается”

Игра актеров в _____ была превосходной.

Какие слова, скорее всего, появятся в этом предложении?

Игра актеров в фильме была превосходной.

Игра актеров в мюзикле была превосходной.

Игра актеров в постановке была превосходной.

“Вы узнаёте слово по компании, которой оно придерживается”

Игра актеров в _____ была превосходной.

Какие слова в этом предложении **вряд ли** появятся?

“Вы узнаёте слово по компании, которой оно придерживается”

Игра актеров в _____ была превосходной.

Какие слова в этом предложении **вряд ли** появятся?

- X Игра актеров в банане была превосходной.
- X Игра актеров в тензоре была превосходной.
- X Игра актеров в утюге была превосходной.

“Вы узнаёте слово по компании, которой оно придерживается”

- Если слова фильм, кино и мюзикл **очень часто** встречаются в одном и том же контексте (то есть в предложениях), то, скорее всего, они связаны между собой.

“Вы узнаёте слово по компании, которой оно придерживается”

- Если слова фильм, кино и мюзикл **очень часто** встречаются в одном и том же контексте (то есть в предложениях), то, скорее всего, они связаны между собой.
- В более общем смысле, родственные слова встречаются в схожих контекстах.

“Вы узнаёте слово по компании, которой оно придерживается”

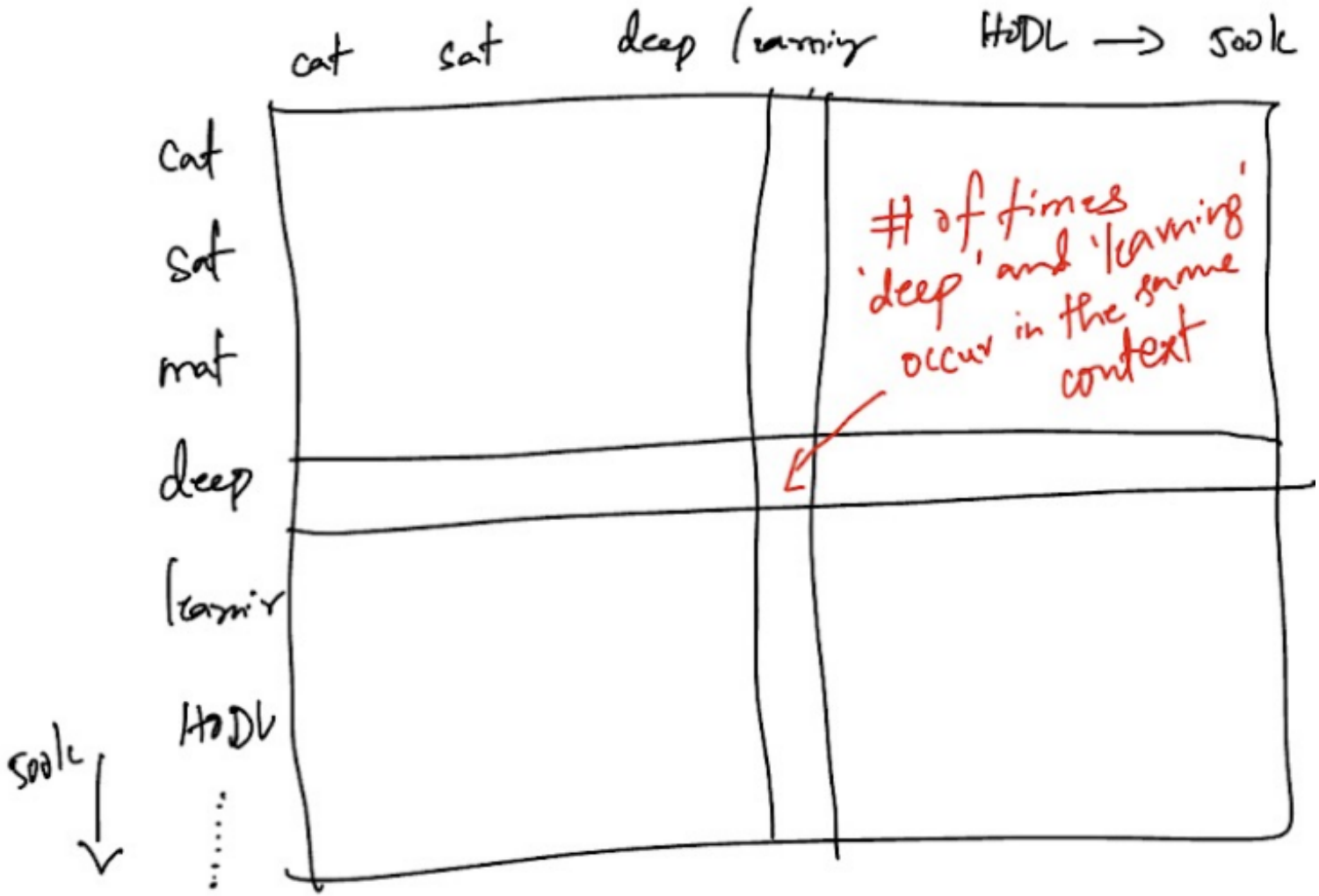
- Если слова фильм, кино и мюзикл **очень часто** встречаются в одном и том же контексте (то есть в предложениях), то, скорее всего, они связаны между собой.
- В более общем смысле, родственные слова встречаются в схожих контекстах.
- Итак, давайте количественно оценим, как часто слова встречаются в схожих контекстах, и попробуем изучить вложение на основе этих данных.

Изучение векторов GloVe(Global Vector) – Интуиция

Представьте, что мы просматриваем каждое предложение в Википедии и делаем следующее:

- Идентифицируем все встречающиеся слова.
- Для каждой пары слов мы подсчитываем, сколько раз они встречаются в одном предложении. В результате получаем матрицу совместной встречаемости слов.

Изучение векторов GloVe – Интуиция



Изучение векторов GloVe – Интуиция

- Если мы сможем узнать(learn) векторы встраивания, которые можно использовать для аппроксимации наблюдаемой матрицы совместной встречаемости, то, скорее всего, эти векторы действительно отражают некоторое понятие семантического расстояния.

Изучение векторов GloVe – Интуиция

- Если мы сможем узнать(learn) векторы встраивания, которые можно использовать для аппроксимации наблюдаемой матрицы совместной встречаемости, то, скорее всего, эти векторы действительно отражают некоторое понятие семантического расстояния.
- Мы можем рассматривать
  - векторы встраивания как просто веса в модели, а
  - матрицу совместной встречаемости как просто данные.

Изучение векторов GloVe – Интуиция

- Если мы сможем узнать(learn) векторы встраивания, которые можно использовать для аппроксимации наблюдаемой матрицы совместной встречаемости, то, скорее всего, эти векторы действительно отражают некоторое понятие семантического расстояния.
- Мы можем рассматривать
  - векторы встраивания как просто веса в модели, а
  - матрицу совместной встречаемости как просто данные.
- Затем мы можем узнать(learn) значения весов, которые минимизируют ошибку прогнозирования

Модель GloVe - обозначения[1]

- Обозначим количество совместных вхождений слова i и слова j как X_{ij}.

[1] смотри детали в https://nlp.stanford.edu/pubs/glove.pdf

Модель GloVe - обозначения[1]

- Обозначим количество совместных вхождений слова i и слова j как X_{ij}.
- Для каждого слова i мы обозначаем вектор встраивания w_i

[1] смотри детали в https://nlp.stanford.edu/pubs/glove.pdf

Модель GloVe - обозначения[1]

- Обозначим количество совместных вхождений слова i и слова j как X_{ij}.
- Для каждого слова i мы обозначаем вектор встраивания w_i
- Каждое слово имеет естественную частоту употребления («movie» против «flick»).
  - Мы хотим, чтобы векторы w_i отражали закономерности совместного появления независимо от собственной частоты.
  - Для определения естественной частоты мы присваиваем каждому слову «смещение» b_i.

[1] смотри детали в https://nlp.stanford.edu/pubs/glove.pdf

GloVe модель.

- Мы предполагаем, что количество совместных появлений пары слов является (простой) линейной функцией двух смещений и двух векторов встраивания следующим образом:

$$X_{ij} = b_i + b_j + w_i^T w_j$$

GloVe модель.

- Мы предполагаем, что количество совместных появлений пары слов является (простой) линейной функцией двух смещений и двух векторов встраивания следующим образом:

$$X_{ij} = b_i + b_j + w_i^T w_j$$

- Однако количество совпадений может сильно варьироваться, поэтому мы можем сузить этот диапазон, используя логарифм этих значений.

$$\log(X_{ij}) = b_i + b_j + w_i^T w_j$$

Решение модели GloVe

$$\log(X_{ij}) = b_i + b_j + w_i^T w_j$$

Как мы можем узнать веса этой модели?

Решение модели GloVe

$$\log(X_{ij}) = b_i + b_j + w_i^T w_j$$

Как мы можем узнать веса этой модели?

$$minimize \sum_{i,j} [\log(X_{ij}) - (b_i + b_j + w_i^T w_j)]^2$$

Решение модели GloVe

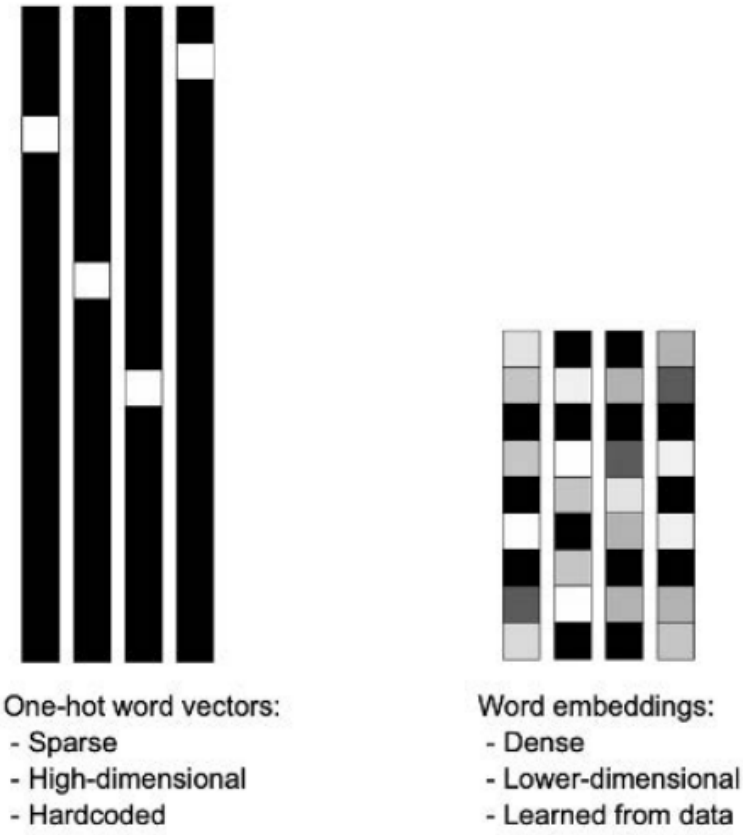
$$\log(X_{ij}) = b_i + b_j + w_i^T w_j$$

Как мы можем узнать веса этой модели?

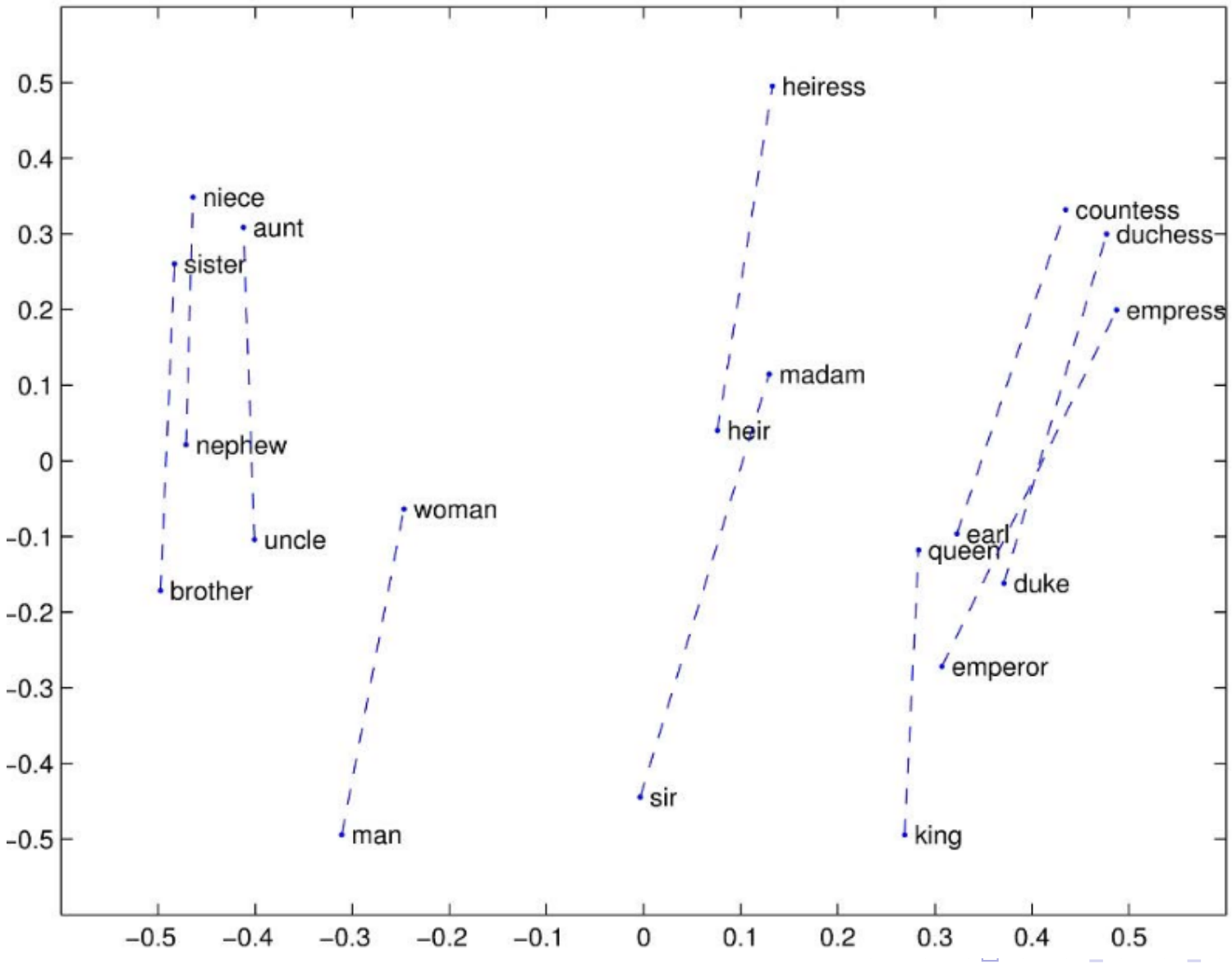
$$minimize \sum_{i,j} [\log(X_{ij}) - (b_i + b_j + w_i^T w_j)]^2$$

Когда мы закончим, мы отбрасываем смещения b и используем только векторы вложения w.

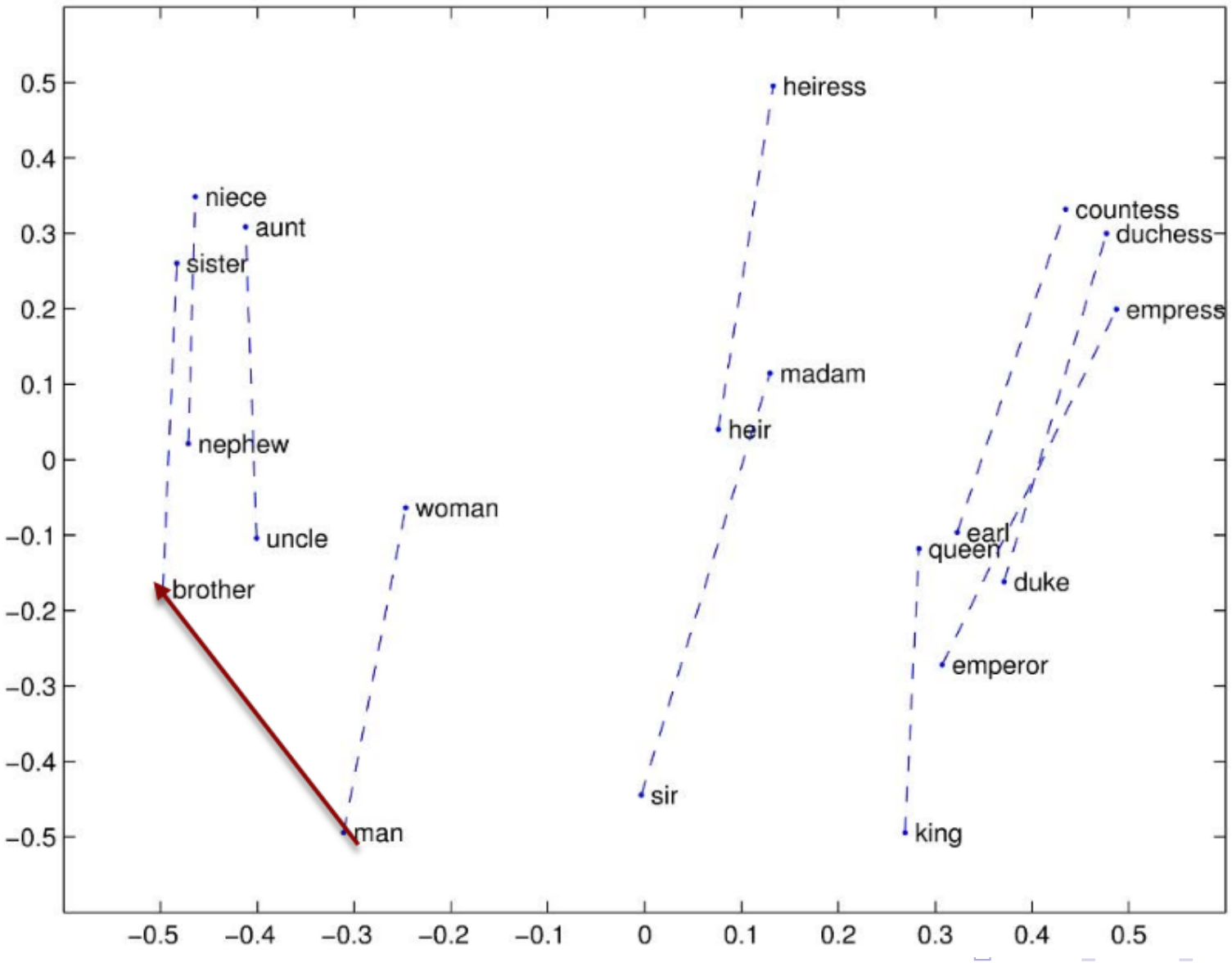
Мы можем выбирать длину этих векторов. Оказывается, векторы встраивания могут быть намного меньше, чем векторы one-hot, потому что они могут быть плотными (в отличие от векторов one-hot, которые по определению являются разреженными).



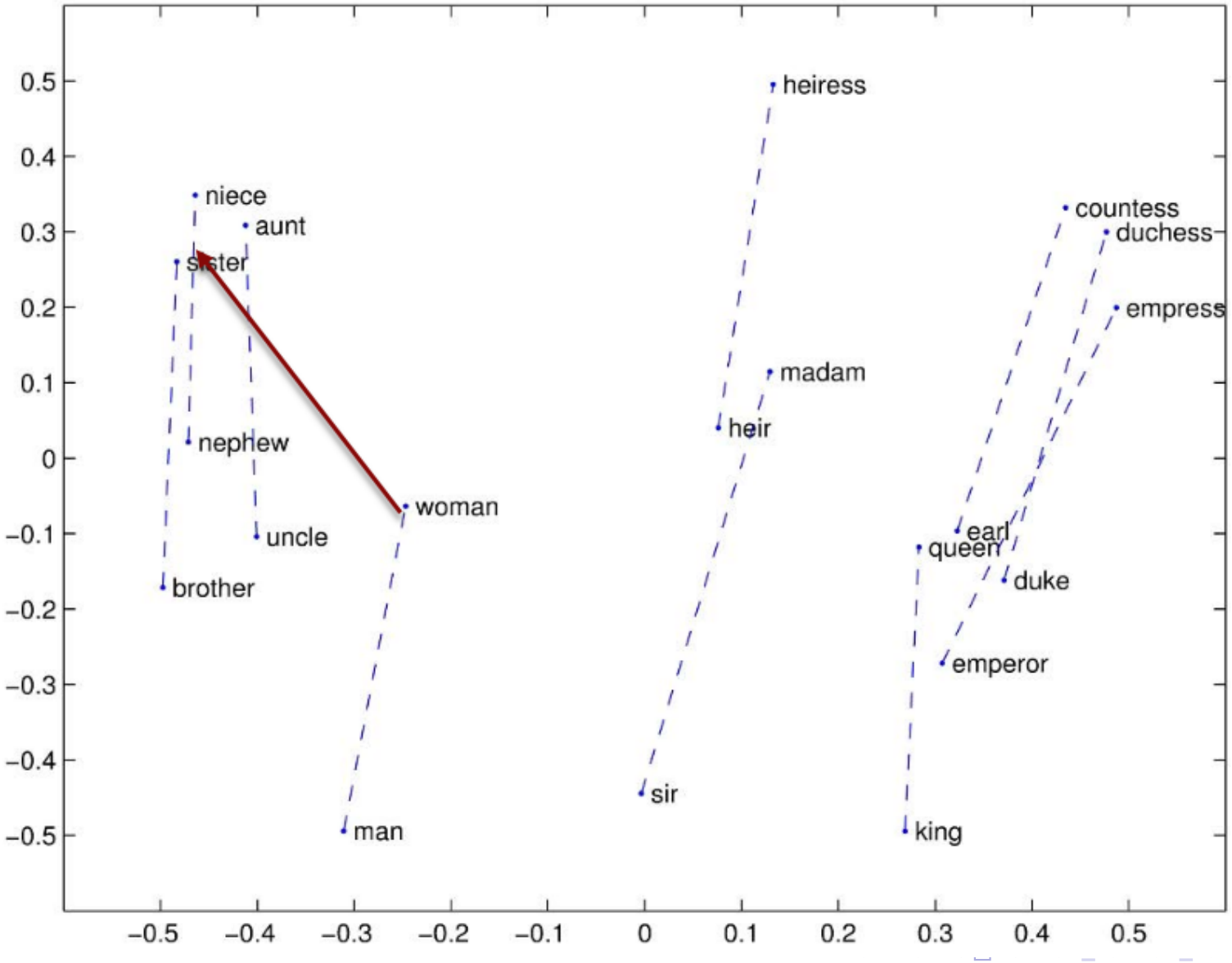
Для векторов GloVe, полученных с помощью этого подхода, семантическое значение действительно отражает геометрическое значение.



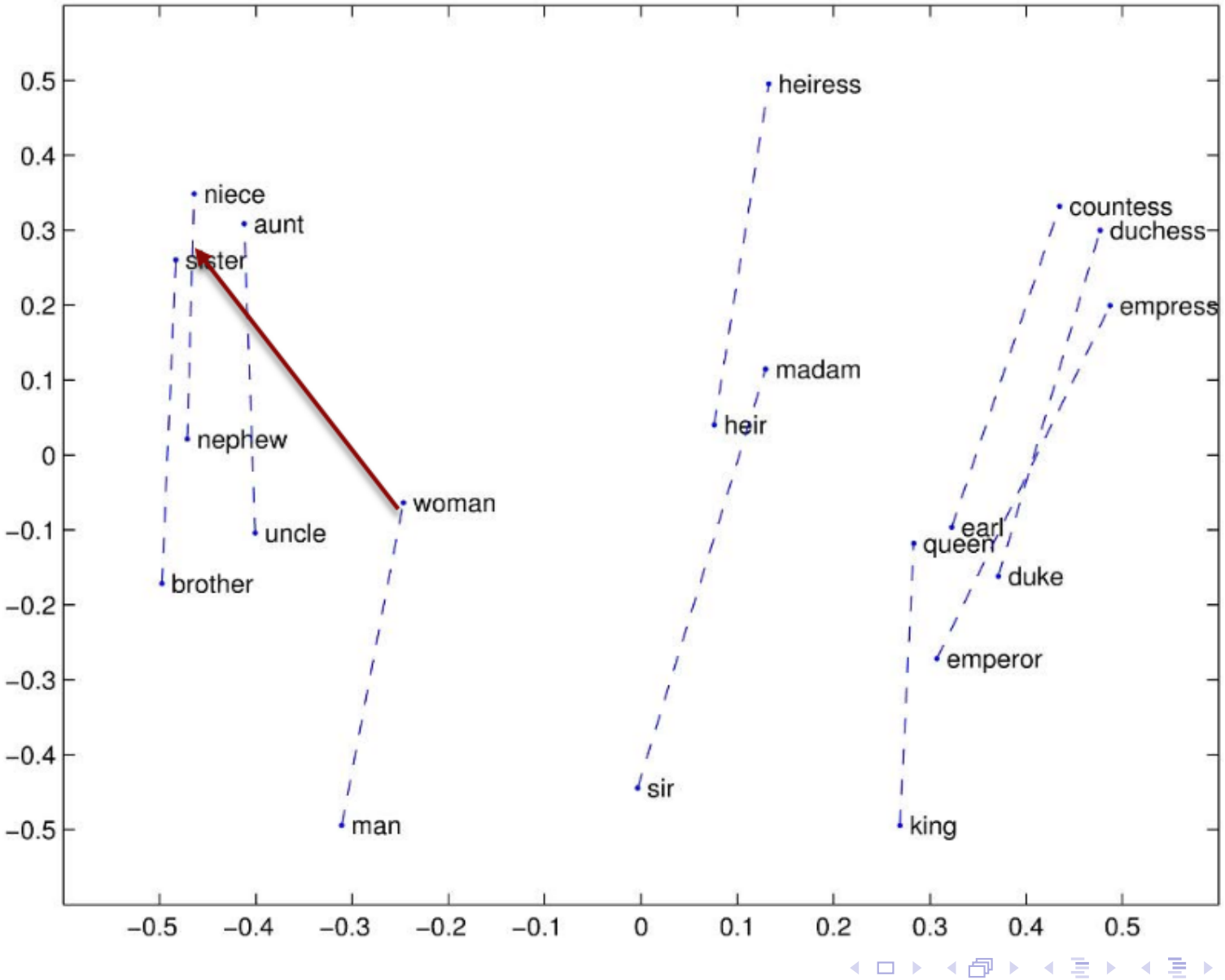
Для векторов GloVe, полученных с помощью этого подхода, семантическое значение действительно отражает геометрическое значение.



Для векторов GloVe, полученных с помощью этого подхода, семантическое значение действительно отражает геометрическое значение.



$$(brother - man) + woman = sister$$



Преимущества и недостатки использования предварительно обученных встраиванний, таких как GloVe.

- Использование предварительно обученных встраиванний (например, GloVe) может быть полезно, если у вас недостаточно данных для обучения вложения вашего словоря для специфической конкретной задачи.

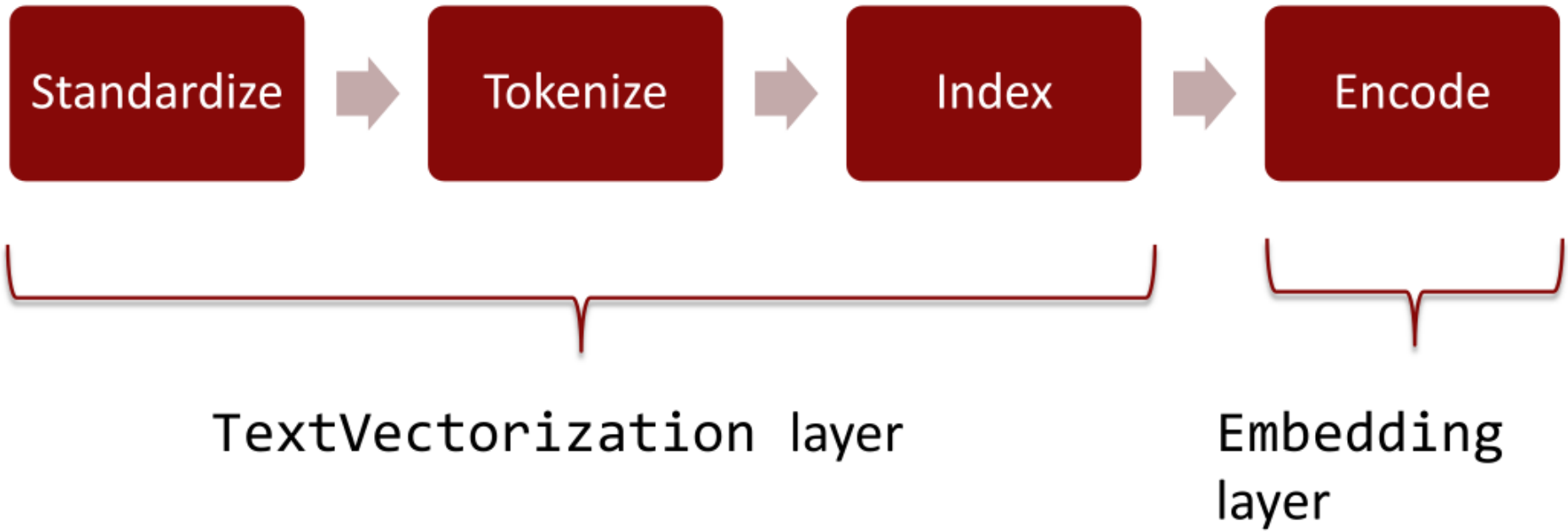
Преимущества и недостатки использования предварительно обученных встраиванний, таких как GloVe.

- Использование предварительно обученных встраиванний (например, GloVe) может быть полезно, если у вас недостаточно данных для обучения вложения вашего словоря для специфической конкретной задачи.
- Недостатком этого подхода является то, что данное векторное представление не будет адаптировано к вашим данным, но оно отражает общие аспекты структуры языка. Это не обязательно плохо, поскольку в большинстве случаев можно ожидать, что характеристики слов будут достаточно общими.

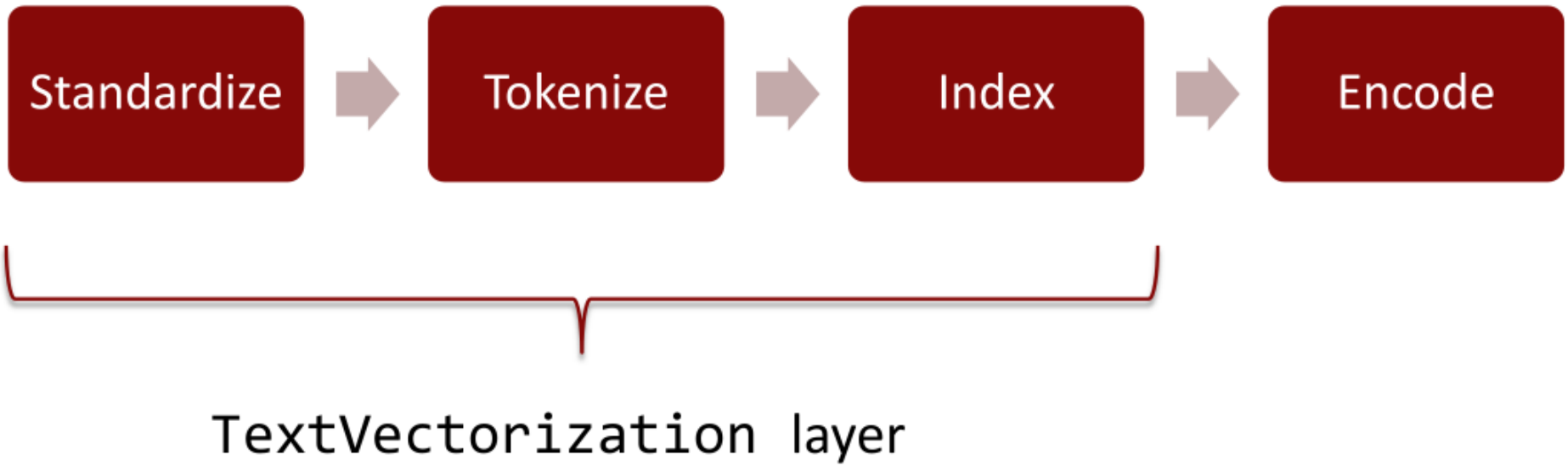
Преимущества и недостатки использования предварительно обученных встраиванний, таких как GloVe.

- Использование предварительно обученных встраиванний (например, GloVe) может быть полезно, если у вас недостаточно данных для обучения вложения вашего словоря для специфической конкретной задачи.
- Недостатком этого подхода является то, что данное векторное представление не будет адаптировано к вашим данным, но оно отражает общие аспекты структуры языка. Это не обязательно плохо, поскольку в большинстве случаев можно ожидать, что характеристики слов будут достаточно общими.
- Мы также можем создавать собственные вложения с нуля. Мы продемонстрируем оба варианта в Colab.

Работа с вложениями в Keras

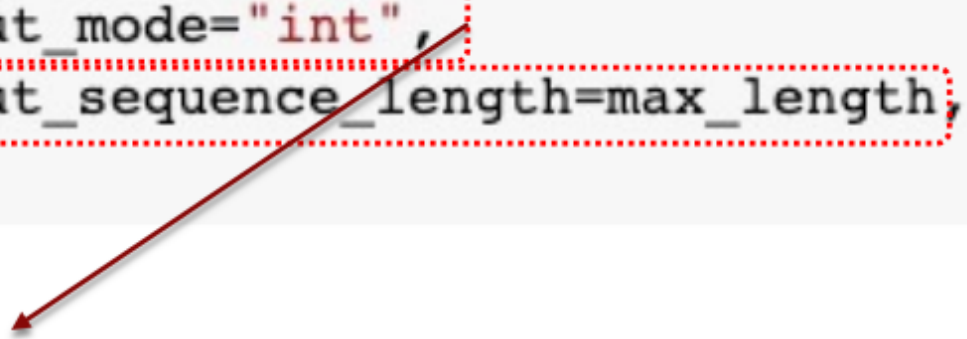


Давайте сначала рассмотрим это.



Два ключевых отличия от прежней ситуации

```
max_length = 300 #90% of songs
max_tokens = 5000

text_vectorization = keras.layers.TextVectorization(
    max_tokens=max_tokens,
    output_mode="int",
    output_sequence_length=max_length,
)
```



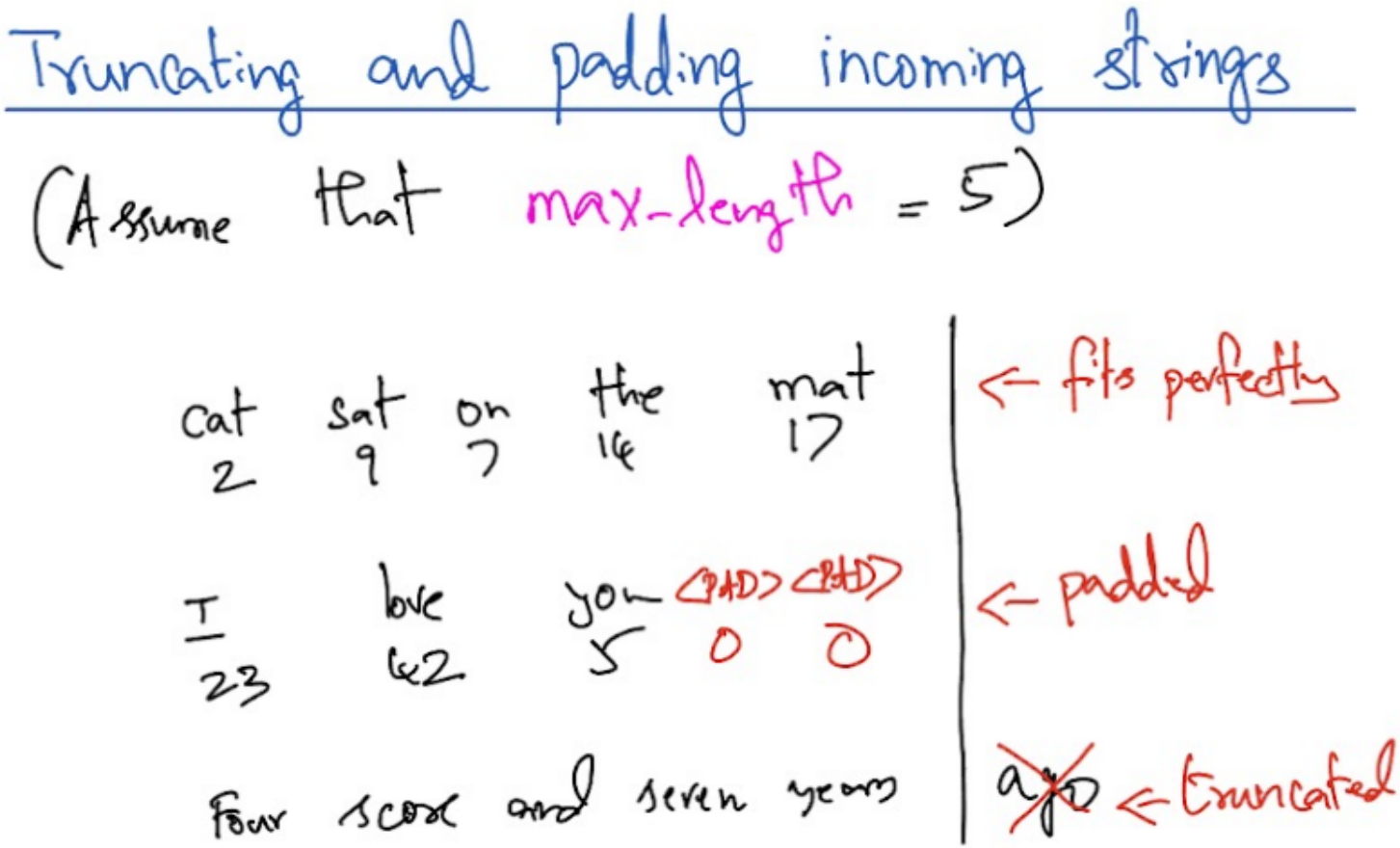
Мы хотим, чтобы слой выполнял только STI-операции, поэтому мы указываем ему остановиться после этапа индексации (т.е., присвоения целого числа каждому токену) и вывести эти целые числа.

Два ключевых отличия от прежней ситуации

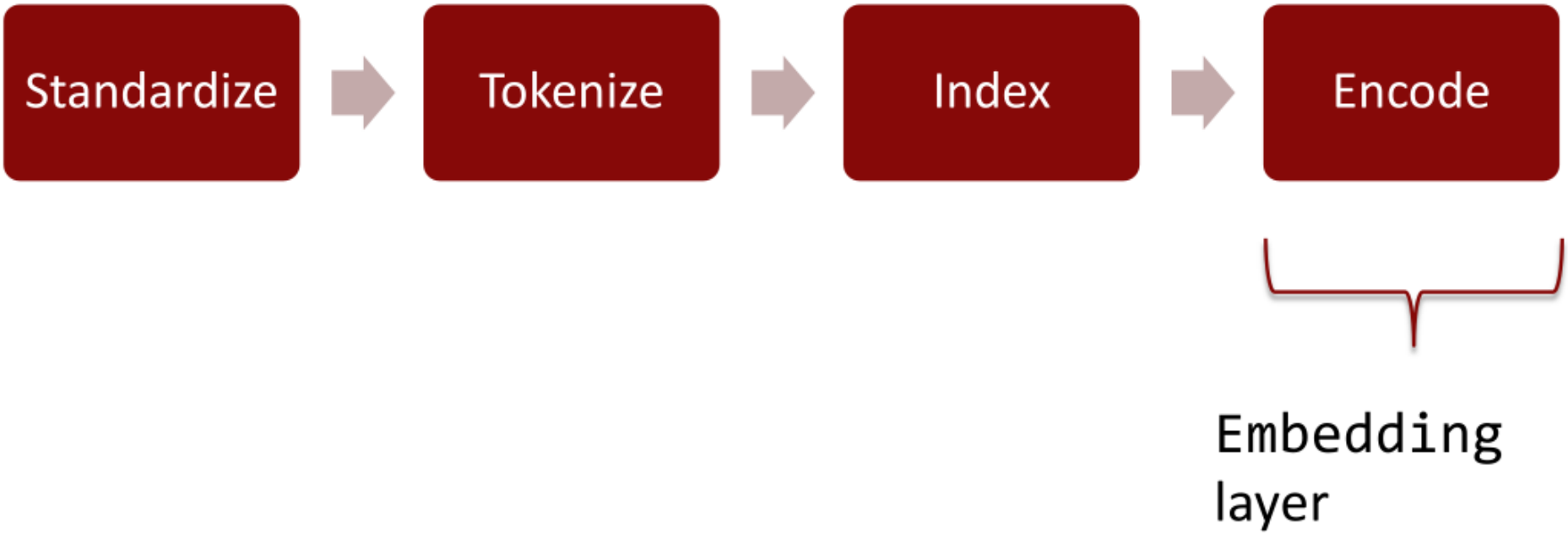
```
max_length = 300 #90% of songs
max_tokens = 5000

text_vectorization = keras.layers.TextVectorization(
    max_tokens=max_tokens,
    output_mode="int",
    output_sequence_length=max_length,
)
```

Поскольку входные предложения имеют разную длину, мы выбираем значение max_length и указываем слою обрезать/дополнить каждое предложение до этой длины (следующий слайд).

Обрезание и дополнение строк

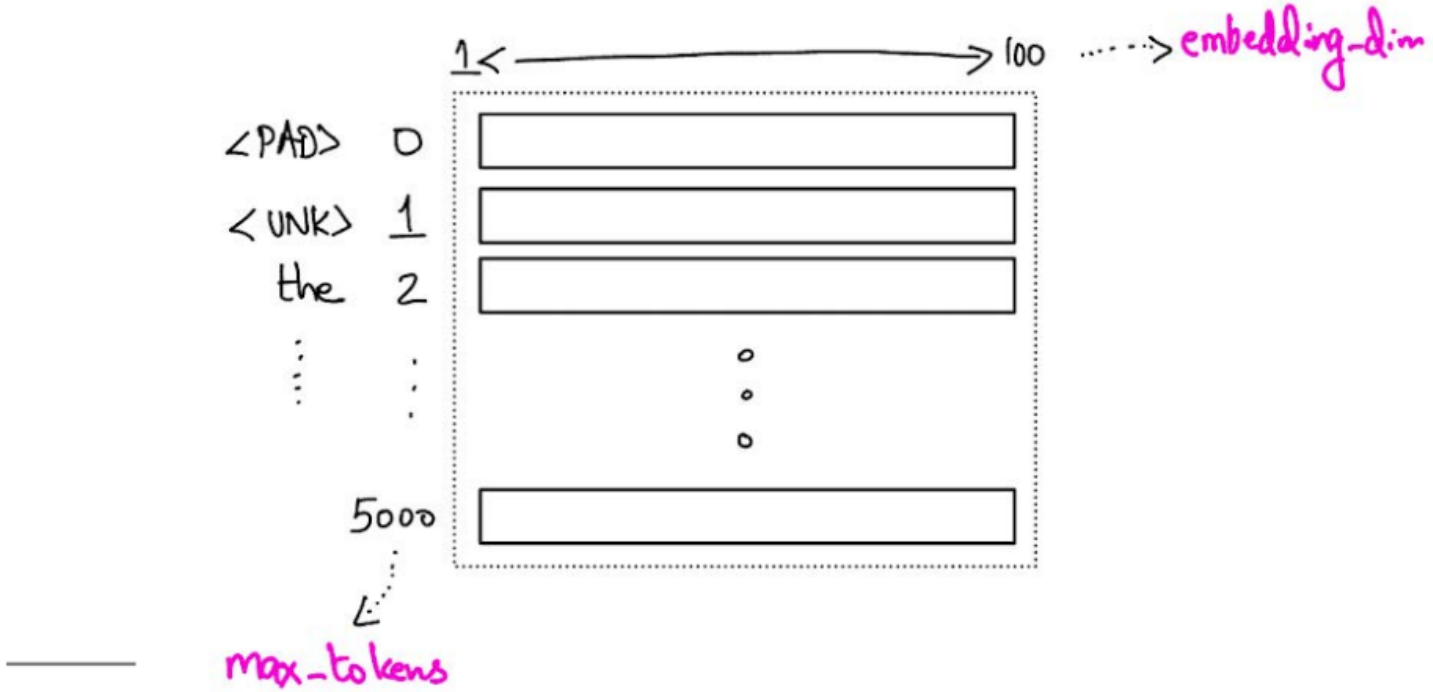


Работа с встраиваниями в Keras



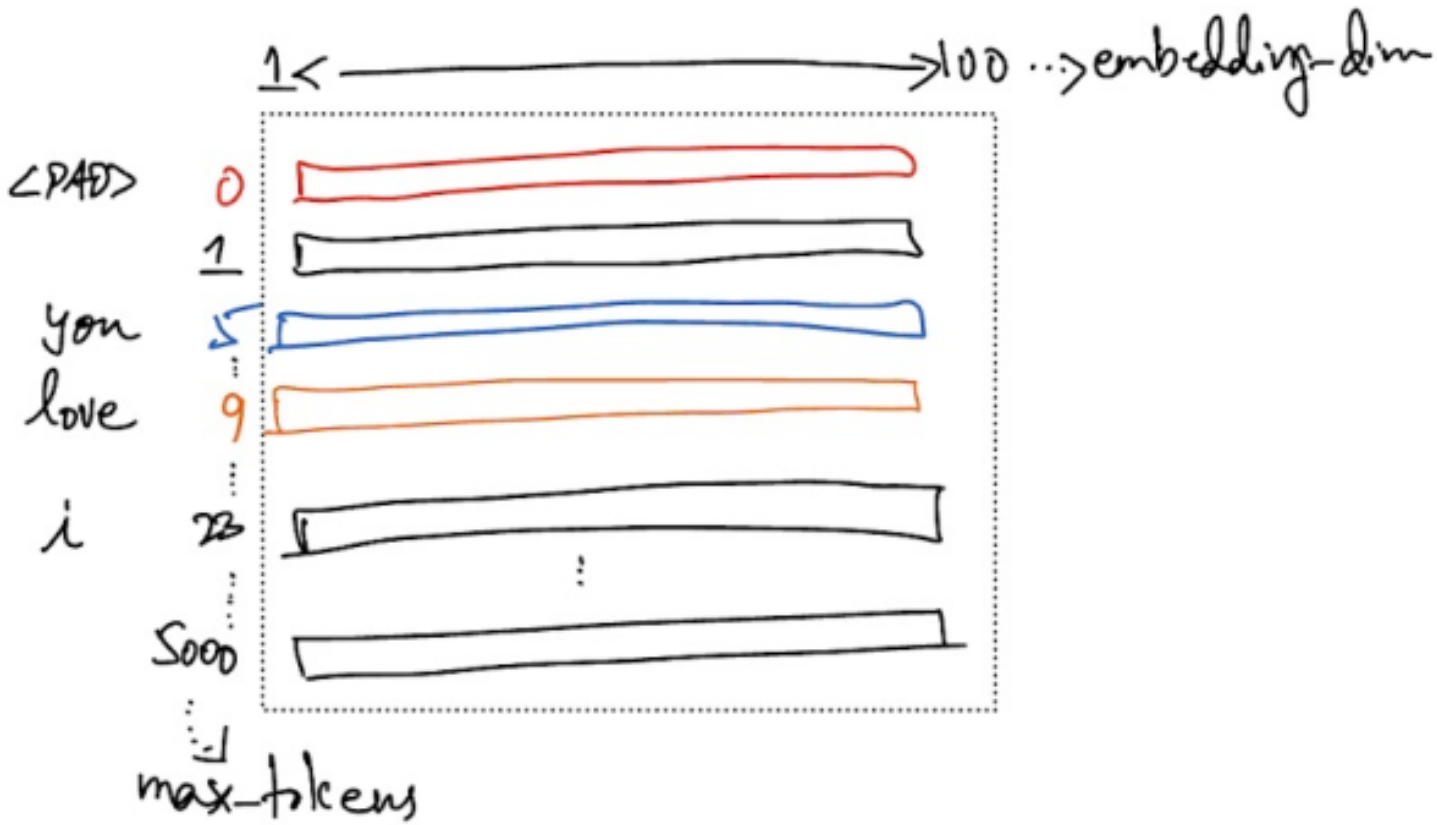
Слой встраивания представляет собой просто таблицу, которая сопоставляет целочисленные индексы с векторами.

```
embedding_dim = 100

keras.layers.Embedding(max_tokens,
                       embedding_dim)
```

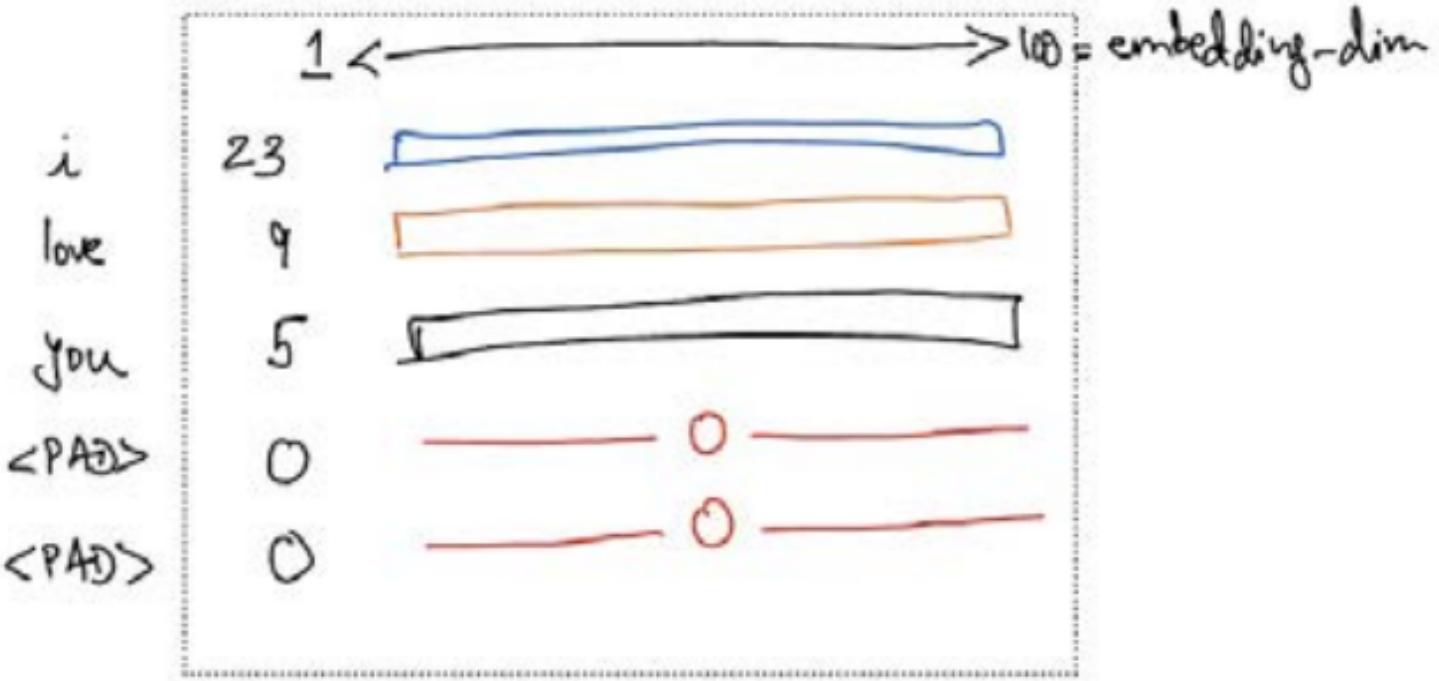


When an input sentence arrives, the TextVectorization layer runs STI and truncates/pads to max_length as needed

The Embedding layer "looks up" the corresponding vectors ...

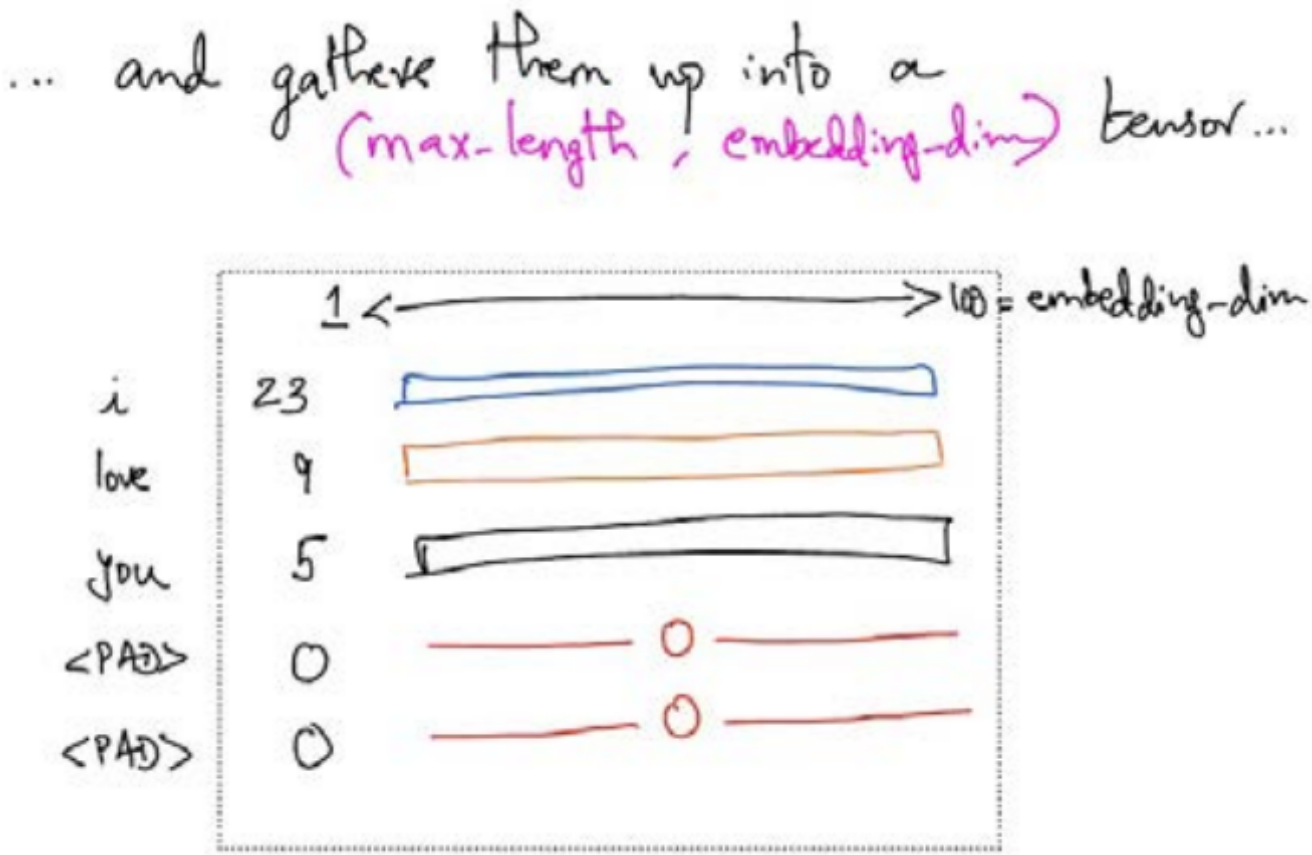


... and gather them up into a (max_length, embedding_dim) tensor...



Эту таблицу необходимо преобразовать в вектор, который можно будет «подать на вход» первого скрытого слоя.

Какие есть варианты?



Эту таблицу необходимо преобразовать в вектор, который можно будет «подать на вход» первого скрытого слоя.

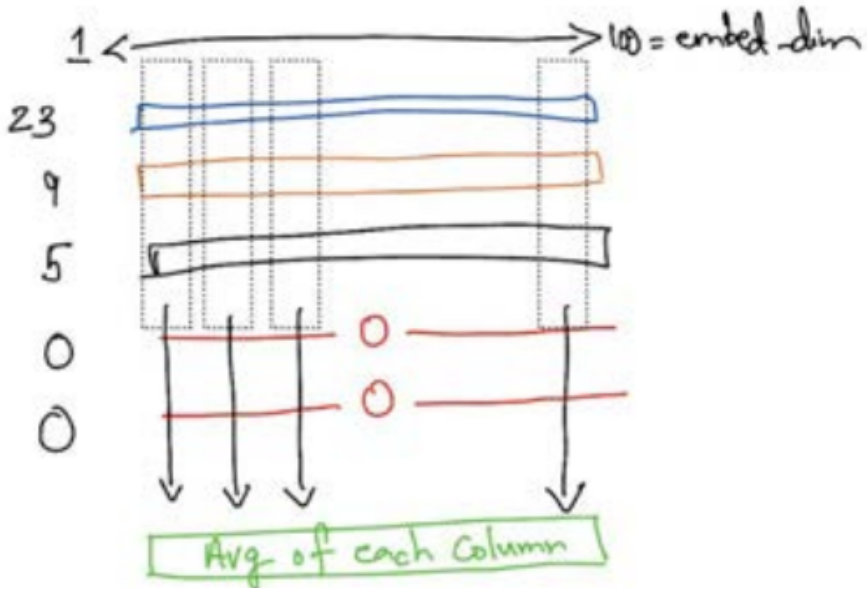
Какие есть варианты?

- Преобразовать в длинный вектор
- Суммировать/усреднить векторы вложения
- ...

Мы усредним их с помощью слоя GlobalAveragePooling1D

```
keras.layers.GlobalAveragePooling1D()
```

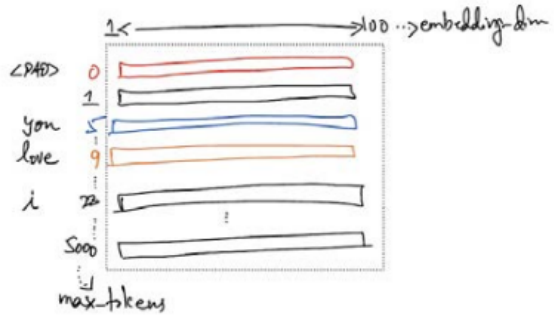
The GlobalAveragePooling1D layer averages each column. This is the vector that will be fed to the first hidden layer



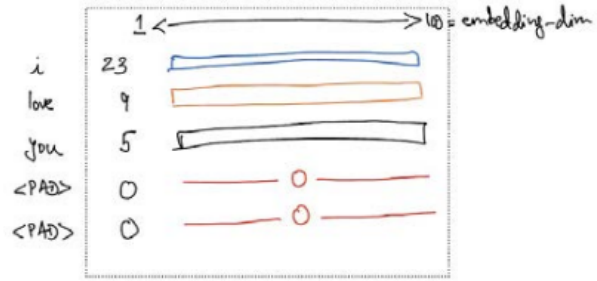
When an input sentence arrives, the TextVectorization layer runs STI and truncates/pads to max_length as needed

"I love you" → STI → i love you PAD PAD
23 9 5 0 0

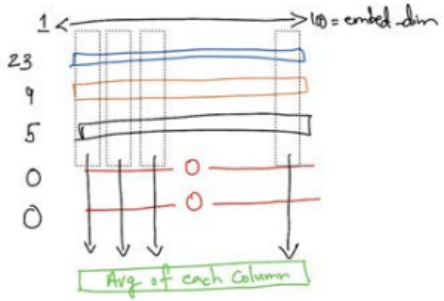
... the Embedding layer "looks up" the corresponding vectors ....



... and gathers them up into a (max_length, embedding_dim) tensor...



The GlobalAveragePooling1D layer averages each column. This is the vector that will be fed to the first hidden layer



Colab

Colab link