

Работа с текстами

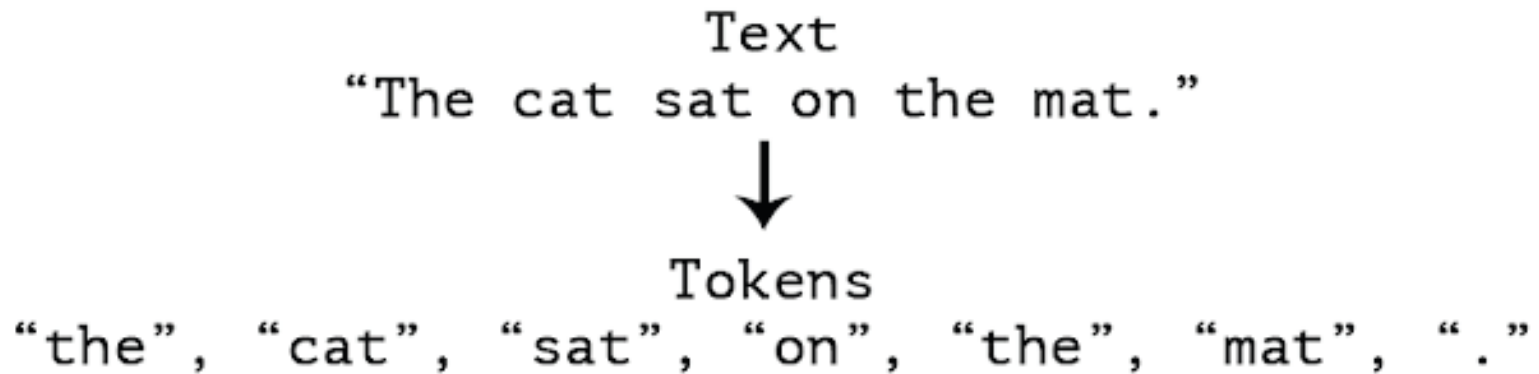
Елена Кантонистова

ТЕРМИНОЛОГИЯ

- документ = текст
- корпус – набор документов
- токен – формальное определение “слова”; токен может не иметь смыслового значения (например, “1 2fdh” или “авыдшл”), но обычно отделен от остальных токенов пробелами или знаками препинания

ТОКЕНИЗАЦИЯ ТЕКСТА

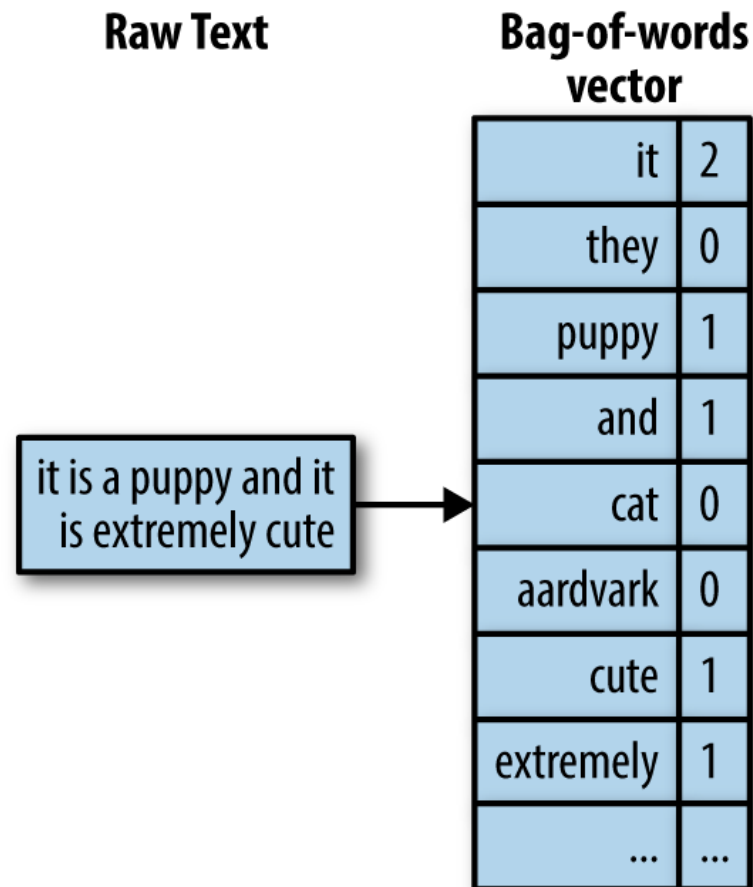
Чтобы работать с текстом, необходимо разбить его на токены. В простейшем случае токены – это слова (а также наборы букв, знаки препинания и т.д.).



МЕТОДЫ КОДИРОВАНИЯ ТЕКСТОВЫХ ДАННЫХ

BAG OF WORDS (МЕШОК СЛОВ)

- По корпусу создадим словарь из всех встречающихся в нем слов (можно убрать общеупотребительные часто встречающиеся слова и очень редкие слова).
- Каждое слово закодируем вектором, в котором стоит единица на месте, соответствующем месту этого слова в словаре, все остальные компоненты вектора – 0.
- Для кодирования документа сложим коды всех его слов.



BAG OF WORDS (ПРИМЕР)

Пусть корпус состоит из следующих документов:

- D1 - “I am feeling very happy today”
- D2 - “I am not well today”
- D3 - “I wish I could go to play”

Кодировка этих документов будет такой:

	I	am	feeling	very	happy	today	not	well	wish	could	go	to	play
D1	1	1	1	1	1	1	0	0	0	0	0	0	0
D2	1	1	0	0	0	1	1	1	0	0	0	0	0
D3	2	0	0	0	0	0	0	0	1	1	1	1	1

BAG OF WORDS

Используя bag of words (BOW), мы теряем информацию о порядке слов в документе.

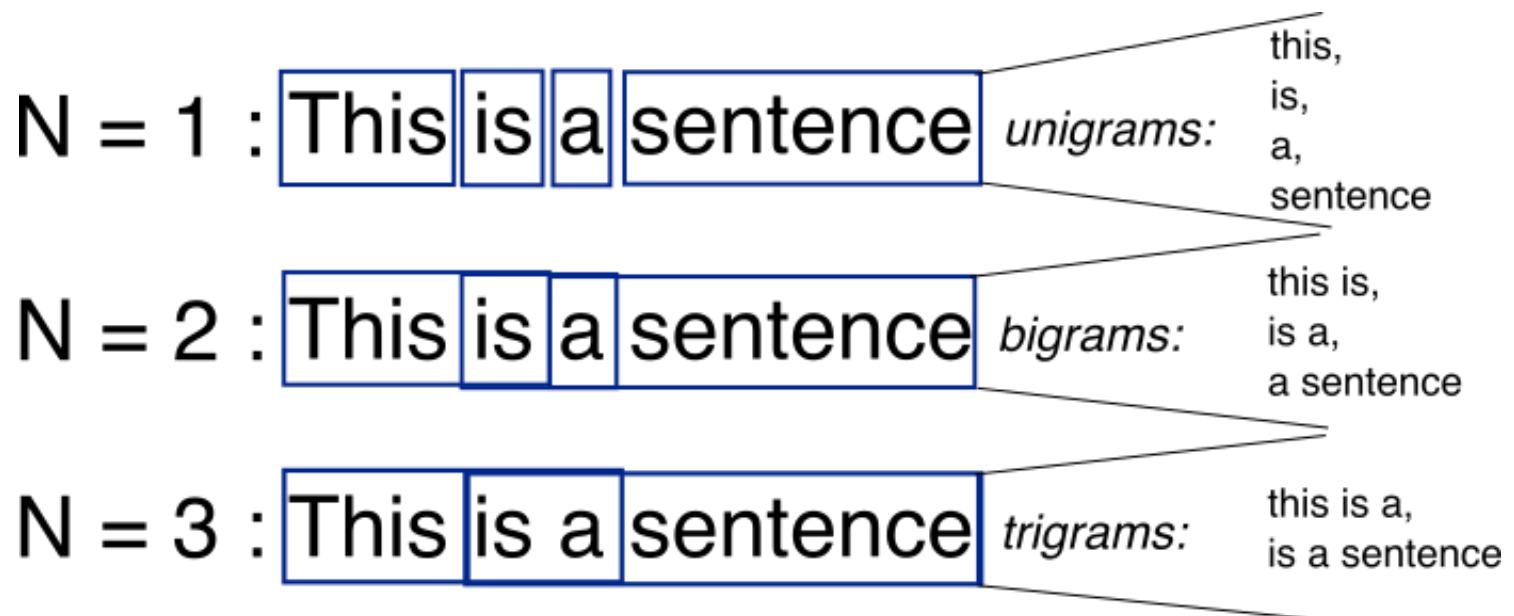
Пример: векторы документов “I have no cats” и “No, I have cats” будут идентичны.

N-GRAM BAG OF WORDS

В качестве слов в словаре можно использовать:

- N-граммы из букв (наборы букв длины N в слове)
- N-граммы из слов (наборы фраз длины N в документе)

Такой подход поможет учесть сходственные слова и опечатки.



TF-IDF

- слова, которые редко встречаются в корпусе, но присутствуют в документе, могут оказаться важными для характеристики документа.
- слова, которые встречаются во всех документах, наоборот, не важны.

TF-IDF

Tf-Idf (term frequency – inverse document frequency):

- *$tf(t, d)$ - частота вхождения слова t в документ d :*

$$tf(t, d) = \frac{n_t}{\sum_k n_k} = \frac{\text{число вхождений слова } t \text{ в документ}}{\text{общее число слов в документе}}$$

$tf(t, d)$ показывает важность слова t в документе d .

TF-IDF

- $tf(t, d)$ - частота вхождения слова t в документ d :

$$tf(t, d) = \frac{n_t}{\sum_k n_k} = \frac{\text{число вхождений слова } \mathbf{t} \text{ в документ}}{\text{общее число слов в документе}}$$

$tf(t, d)$ показывает важность слова t в документе d .

- $idf(t, D)$ - величина, обратная частоте, с которой слово t встречается в документах корпуса D .

$$idf(t, D) = \log \frac{|D|}{|\{d_i \in D \mid t \in d_i\}|},$$

$|D|$ — число документов в корпусе,

$|\{d_i \in D \mid t \in d_i\}|$ - число документов, в которых встречается слово t

Учёт idf уменьшает вес часто используемых в корпусе слов.

TF-IDF

Tf-idf слова t в документе d из корпуса D :

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D),$$

Пример:

Дана коллекция D из $10000000 = 10^7$ документов, в 1000 из них встречается слово “заяц”. В данном документе d из коллекции 100 слов, и слово “заяц” встречается 3 раза.

$$tf(\text{заяц}, d) = \frac{3}{100} = 0,03$$

$$idf(\text{заяц}, D) = \log\left(\frac{10^7}{10^3}\right) = 4$$

Поэтому $tfidf(\text{заяц}, d, D) = 0,03 \cdot 4 = 0,12$.