Кейс стади 2

Обработка текста: bag of words. Теорема Байеса и наивный байесовский классификатор. Анализ временного ряда. Кластеризация: k-means, EM-алгоритм. Определение тональности текста. Определение спама в тексте. Тестирование гипотез (статистические тесты на нескольких выборках).

Юстина Иванова

Специалист по Анализу Данных

Инженер-программист МГТУ им. Баумана

Master of Science in Artificial Intelligence
University of Southampton

Специалист по анализу данных
в компании ОЦРВ

Юстина Иванова
студент-аспирант
University of Bolzano

НЕТОЛОГИЯ

Простейший спам-фильтр

(использовались года до 2010)

привет... 1829
валера ...1710
нет ... 1191
куда ... 1012
небо ...985
огурцы ... 873
говорить...747
третий ... 739

нормальные письма

672 раза

«котик»

13 раз

виагра ... 1552
казино ... 1492
100% ... 1320
кредит... 1184
скидка ... 985
нажми ... 873
free ... 747
доход ... 739

спам-письма

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

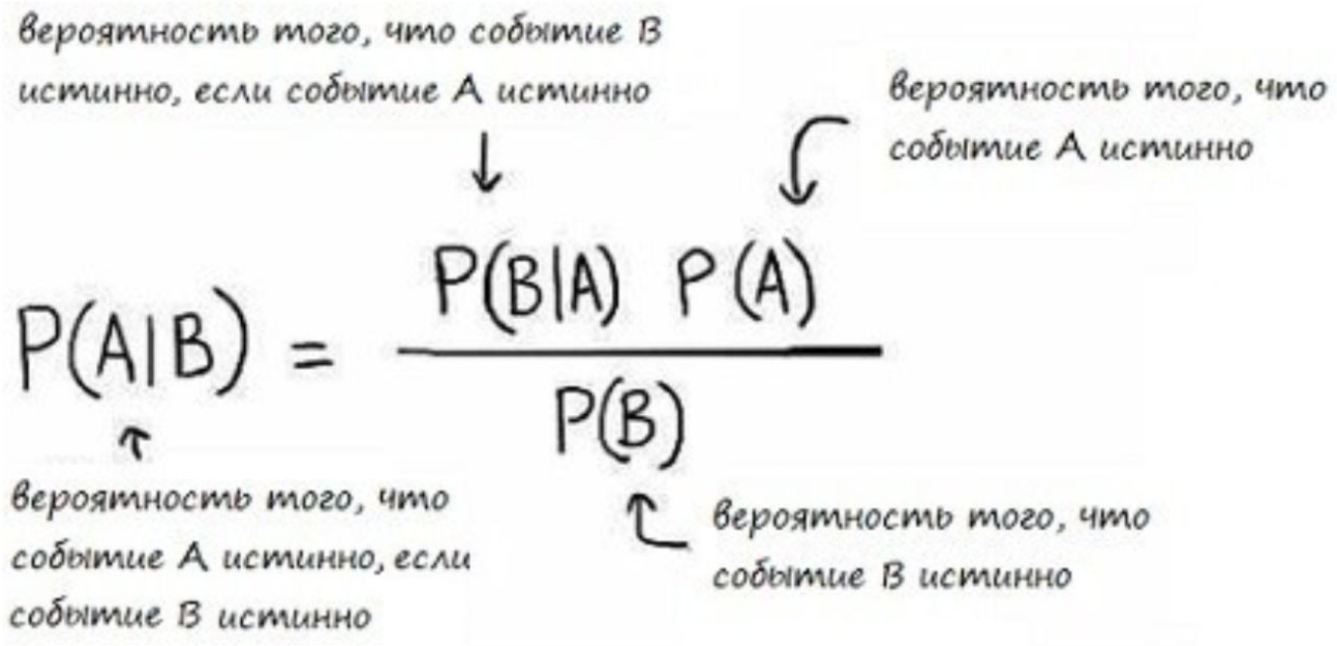
формула Байеса

не спам

Наивный Байес

https://vas3k.ru/blog/machine_learning/

Формула Байеса



http://baguzin.ru/wp/den-morris-teorema-bajesa-vizualnoe-vvedenie-dlya-nachinayushhih/

Bag of words

The Bag of Words Representation

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!

it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
…	…

15

https://en.wikipedia.org/wiki/Bag-of-words_model

Bag of words

Несмотря на потерю информации о порядке слов в тексте, можно вычислять расстояние между векторами, например, с помощью косинусной метрики.
Мы можем пойти дальше и представить наш корпус (набор текстов) в виде матрицы “слово-документ” (term-document).

$$
\begin{array}{c} X \\ (\mathbf{d}_j) \\ \downarrow \end{array}
\qquad
U \qquad \Sigma \qquad
\begin{array}{c} V^T \\ (\hat{\mathbf{d}}_j) \\ \downarrow \end{array}
$$

$$
(\mathbf{t}_i^T) \rightarrow
\begin{bmatrix} x_{1,1} & \dots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \dots & x_{m,n} \end{bmatrix}
=
(\hat{\mathbf{t}}_i^T) \rightarrow
\begin{bmatrix} \begin{bmatrix} \\ \mathbf{u}_1 \\ \\ \end{bmatrix} & \dots & \begin{bmatrix} \\ \mathbf{u}_l \\ \\ \end{bmatrix} \end{bmatrix}
\cdot
\begin{bmatrix} \sigma_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_l \end{bmatrix}
\cdot
\begin{bmatrix} [& \mathbf{v}_1 &] \\ & \vdots & \\ [& \mathbf{v}_l &] \end{bmatrix}
$$

https://habr.com/ru/company/ods/blog/329410/

TF-IDF: term frequency — inverse document frequency

$$TF - IDF(w, d, C) = \frac{count(w, d)}{count(d)} * log(\frac{\sum_{d' \in C} 1(w, d')}{|C|})$$

Итак, TF — это частота слова w в тексте d, здесь нет ничего сложного.
А вот IDF — существенно более интересная вещь: это логарифм обратной частоты распространенности слова w в корпусе C. Распространенностью называется отношение числа текстов, в которых встретилось искомое слово, к общему числу текстов в корпусе. С помощью TF-IDF тексты также можно сравнивать, и делать это можно с меньшей опаской, чем при использовании обычных частот.

https://habr.com/ru/company/ods/blog/329410/

Спасибо за внимание!