



# Наивный байесовский классификатор. Фильтрация спама

- 
- 
- **Наивный байесовский классификатор** — простой вероятностный классификатор, основанный на применении теоремы Байеса со строгими (наивными) предположениями о независимости.
  - **Теорема Байеса (или формула Байеса)** — одна из основных теорем элементарной теории вероятностей, которая позволяет определить вероятность какого-либо события при условии, что произошло другое статистически взаимозависимое с ним событие
  - **Формула Байеса** может быть выведена из основных аксиом теории вероятностей, в частности из условной вероятности.



## *Формула Байеса:*

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

где  $P(A)$  — априорная вероятность гипотезы  $A$  ;

$P(A|B)$  — вероятность гипотезы  $A$  при наступлении события  $B$  (апостериорная вероятность);

$P(B|A)$  — вероятность наступления события  $B$  при истинности гипотезы  $A$ ;

$P(B)$  — полная вероятность наступления события  $B$ .

# ***Условная вероятность***

Это вероятность наступления события А при условии, что событие В произошло.

Обозначение вероятности:

$$P(A|B)$$

Очевидный частный случай:

$$P(A|A)=1=100\%$$

Вероятность совместного появления двух зависимых событий равна:

$$P(AB)= P(A|B) P(B)= P(B|A)P(A)$$



## *Доказательство*

$$P(AB) = P(A|B)P(B) = P(B|A)P(A)$$

## *Следовательно*

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

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

**Задача.** Состоится ли матч при солнечной погоде (sunny)?  
Мы можем решить эту задачу с помощью описанного выше подхода.

$$P(\text{Yes} \mid \text{Sunny}) = (P(\text{Sunny} \mid \text{Yes}) * P(\text{Yes})) / P(\text{Sunny})$$

Здесь мы имеем следующие значения:

$$P(\text{Sunny} \mid \text{Yes}) = 3 / 9 = 0,33$$

$$P(\text{Sunny}) = 5 / 14 = 0,36$$

$$P(\text{Yes}) = 9 / 14 = 0,64$$

Теперь рассчитаем  $P(\text{Yes} \mid \text{Sunny})$ :

$$P(\text{Yes} \mid \text{Sunny}) = 0,33 * 0,64 / 0,36 = 0,60$$

Значит, при солнечной погоде более вероятно, что матч состоится.

# Модель наивного байесовского классификатора

Вероятностная модель для классификатора — это условная модель

$$p(C|F_1, \dots, F_n)$$

над зависимой переменной класса  $C$  с малым количеством результатов или классов, зависящая от нескольких переменных  $F_1, \dots, F_n$ .

Используя теорему Байеса, запишем

$$p(C | F_1, \dots, F_n) = \frac{p(C) p(F_1, \dots, F_n | C)}{p(F_1, \dots, F_n)}.$$



Числитель эквивалентен совместной вероятности модели

$$p(C, F_1, \dots, F_n)$$


$$\begin{aligned} p(C, F_1, \dots, F_n) &= p(C)p(F_1, \dots, F_n|C) = p(C)p(F_1|C)p(F_2, \dots, F_n|C, F_1) \\ &= p(C)p(F_1|C) p(F_2|C, F_1)p(F_3, \dots, F_n|C, F_1, F_2) \\ &= p(C)p(F_1|C) p(F_2|C, F_1) \dots p(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}) \end{aligned}$$

каждое свойство  $F_i$  условно независимо от любого другого свойства  $F_j$  при  $j \neq i$ . Это означает:

$$p(F_i|C, F_j) = p(F_i|C)$$

$$\begin{aligned} p(C, F_1, \dots, F_n) &= p(C)p(F_1|C) p(F_2|C)p(F_3|C) \dots p(F_n|C) = \\ &= p(C) \prod_{i=1}^n p(F_i|C) \end{aligned}$$





Условное распределение по классовой переменной  $C$  может быть выражено так:

$$p(C, F_1, \dots, F_n) = \frac{1}{Z} p(C) \prod_{i=1}^n p(F_i|C)$$

где  $Z = p(F_1, \dots, F_n)$  — это масштабный множитель, зависящий только от  $F_1, \dots, F_n$ , то есть константа, если значения переменных известны.



# *Оценка параметров*

Непрерывные свойства оценок максимального правдоподобия вероятностей, как правило, оцениваются через нормальное распределение. В качестве математического ожидания и дисперсии вычисляются статистики — среднее арифметическое и среднеквадратическое отклонение соответственно.

Если данный класс и значение свойства никогда не встречаются вместе в наборе обучения, тогда оценка, основанная на вероятностях, будет равна нулю. Это проблема, так как при перемножении нулевая оценка приведет к потере информации о других вероятностях. Поэтому предпочтительно проводить небольшие поправки во все оценки вероятностей так, чтобы никакая вероятность не была строго равна нулю.

## *Построение классификатора по вероятностной модели*

*Апостериорное правило принятия решения (MAP).*  
Соответствующий классификатор — это функция *classify*, определённая следующим образом:

$$\text{classify}(f_1, \dots, f_n) = \arg \max_c p(C = c) \prod_{i=1}^n p(F_i = f_i | C = c)$$

**Байесовская фильтрация спама** — метод для фильтрации спама, основанный на применении наивного байесовского классификатора, опирающегося на прямое использование теоремы Байеса.

# *История*

Первой известной программой, фильтрующей почту с использованием байесовского классификатора, была программа iFile Джейсона Ренни, выпущенная в 1996 году.

Первая академическая публикация по наивной байесовской фильтрации спама появилась в 1998 году.

В 2002 г. Пол Грэм смог значительно уменьшить число ложноположительных срабатываний.



## Описание

При обучении фильтра для каждого встреченного в письмах слова высчитывается и сохраняется его «вес» — оценка вероятности того, что письмо с этим словом — спам.

При проверке вновь пришедшего письма вероятность «спамовости» вычисляется по формуле

$$P(B) = \sum_{i=1}^N P(A_i)P(B|A_i)$$

В данном случае «гипотезы» — это слова, и для каждого слова «достоверность гипотезы»  $P(A_i) = N_{word\_i} / N_{words\_total}$  — доля этого слова в письме, а «зависимость события от гипотезы»  $P(B|A_i)$  — вычисленный ранее «вес» слова.



# *Математические основы*

Почтовые байесовские фильтры основываются на теореме Байеса. Теорема Байеса используется несколько раз в контексте спама:

- в первый раз, чтобы вычислить вероятность, что сообщение — спам, зная, что данное слово появляется в этом сообщении;
- во второй раз, чтобы вычислить вероятность, что сообщение — спам, учитывая все его слова (или соответствующие их подмножества);
- иногда в третий раз, когда встречаются сообщения с редкими словами.



## *Вычисление вероятности того, что сообщение, содержащее данное слово, является спамом*

$$\Pr(S|W) = \frac{\Pr(W|S) * \Pr(S)}{\Pr(W)} = \frac{\Pr(W|S) * \Pr(S)}{\Pr(W|S) * \Pr(S) + \Pr(W|H) * \Pr(H)}$$

где:

$\Pr(S|W)$  — условная вероятность того, что сообщение—спам, при условии, что слово  $W$ =«replica» находится в нём;

$\Pr(S)$  — полная вероятность того, что произвольное сообщение—спам;

$\Pr(W|S)$  — условная вероятность того, что слово «replica» появляется в сообщениях, если они являются спамом;

$\Pr(H)$  — полная вероятность того, что произвольное сообщение не спам (то есть «ham»);

$\Pr(W|H)$  — условная вероятность того, что слово «replica» появляется в сообщениях, если они являются «ham».



## Спамовость слова

$$\Pr(S|W) = \frac{\Pr(W|S)}{\Pr(W|S) + \Pr(W|H)}$$

Значение  $\Pr(S|W)$  называют «спамовостью» слова  $W$ ;

Значение  $\Pr(W|S)$  приближённо равно относительной частоте сообщений, которые содержат слово  $W$  в сообщениях

$$\Pr(W_i|S) = \frac{\text{count}(M: W_i \in M, M \in S)}{\sum_j \text{count}(M: W_j \in M, M \in S)}$$


$\Pr(W|H)$  приближённо равно относительной частоте сообщений, содержащих слово  $W$  в сообщениях

$$\Pr(W_i|H) = \frac{\text{count}(M: W_i \in M, M \in H)}{\sum_j \text{count}(M: W_j \in M, M \in H)}$$

## Объединение индивидуальных вероятностей

Исходя из «наивного» предположения, для решения задачи классификации сообщений лишь на 2 класса:  $S$  (спам) и  $H = \neg S$  («хэм», то есть не спам) из теоремы Байеса можно вывести следующую формулу оценки вероятности «спамовости» всего сообщения, содержащего слова  $W_1, W_2, \dots W_N$

$$\begin{aligned} p(S|W_1, W_2, \dots W_N) &= [\text{по теореме Байеса}] \\ &= \frac{p(W_1, W_2, \dots W_N|S) * p(S)}{p(W_1, W_2, \dots W_N)} = \\ &= [\text{так как } W_i \text{ предполагаются независимыми}] = \\ &= \frac{\prod_i p(W_i|S) * p(S)}{p(W_1, W_2, \dots W_N)} = \\ &= [\text{по теореме Байеса}] \end{aligned}$$



$$= \frac{\prod_i \frac{p(S|W_i) * p(W_i)}{p(S)} * p(S)}{p(W_1, W_2, \dots W_N)} =$$

= [по формуле полной вероятности] =

$$= \frac{\prod_i \frac{p(S|W_i) * p(W_i)}{p(S)} * p(S)}{\prod_i (p(W_i|S)) * p(S) + \prod_i (p(W_i|\neg S)) * p(\neg S)} =$$

$$= \frac{\prod_i (p(S|W_i) * p(W_i)) * p(S)^{1-N}}{\prod_i (p(S|W_i) * p(W_i)) * p(S)^{1-N} + \prod_i (p(\neg S|W_i) * p(W_i)) * p(\neg S)^{1-N}}$$

$$=$$

$$\frac{\prod_i p(S|W_i)}{\prod_i (p(S|W_i)) + \left(\frac{p(\neg S)}{p(S)}\right)^{1-N} * \prod_i p(\neg S|W_i)}$$



Таким образом, предполагая  $p(S) = p(\neg S) = 0.5$ , имеем:

$$p = \frac{p_1 p_2 \dots p_N}{p_1 p_2 \dots p_N + (1 - p_1)(1 - p_2) \dots (1 - p_N)}$$

где:

$p = \Pr(S \mid W_1, W_2, \dots, W_N)$  — вероятность, что сообщение, содержащее слова  $W_1, W_2, \dots, W_N$  — спам;

$p_1$  — условная вероятность  $p(S \mid W_1)$  того, что сообщение — спам, при условии, что оно содержит первое слово (к примеру, «replica»);

$p_2$  — условная вероятность  $p(S \mid W_2)$  того, что сообщение — спам, при условии, что оно содержит второе слово (к примеру, «watches»);

$p_N$  — условная вероятность  $p(S \mid W_N)$  того, что сообщение — спам, при условии, что оно содержит N-е слово (к примеру, «home»).



## *Проблема редких слов*

Она возникает в случае, если слово никогда не встречалось во время фазы обучения: и числитель, и знаменатель равны нулю, и в общей формуле, и в формуле спамовости.

В целом, слова, с которыми программа столкнулась только несколько раз во время фазы обучения, не являются репрезентативными. Простое решение состоит в том, чтобы игнорировать такие ненадёжные слова.



## *Другие эвристические усовершенствования*

«Нейтральные» слова — такие, как, «the», «a», «some», или «is» (в английском языке), или их эквиваленты на других языках — могут быть проигнорированы. Вообще говоря, некоторые байесовские фильтры просто игнорируют все слова, у которых спамовость около 0.5, так как в этом случае получается качественно лучшее решение. Учитываются только те слова, спамовость которых около 0.0 (отличительный признак законных сообщений — «ham»), или рядом с 1.0 (отличительный признаки спама).





# *Характеристика*

## ПЛЮСЫ

- Прост
- Удобен
- Эффективен

## МИНУСЫ

- Базируется на предположении, что одни слова чаще встречаются в спаме, а другие — в обычных письмах, и неэффективен, если данное предположение неверно
- Работает только с текстом