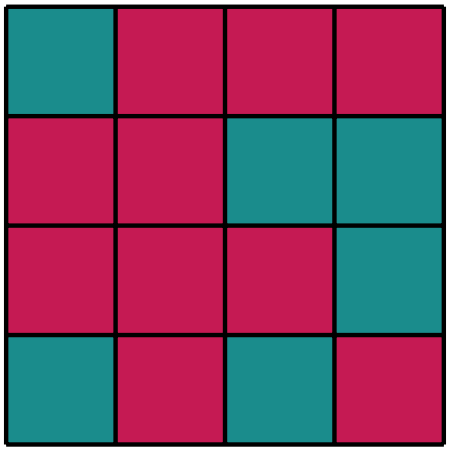
Procesamiento de Lenguaje Natural

Olivia Gutú y Julio Waissman

Maestría en Ciencia de Datos
Semana 2: Análisis de sentimientos con Naïve Bayes

Introducción

Corpus de N tuits



$$P(pos) = \frac{N_{pos}}{N} = \frac{6}{16}$$
$$P(neg) = 1 - P(pos) = \frac{10}{16}$$

$$P(feliz) = \frac{3}{16}$$
$$P(pos \cap feliz) = \frac{1}{16}$$

Regla de Bayes

Probabilidades condicionales:

$$P(pos|feliz) = \frac{1}{3} \qquad P(feliz|pos) = \frac{1}{6}$$

$$\downarrow \qquad\qquad \downarrow$$

$$\frac{P(pos \cap feliz)}{P(feliz)} \qquad \frac{P(feliz \cap\ pos)}{P(pos)}$$

por tanto:

$$P(pos|feliz)P(feliz) = P(feliz|pos)P(pos)$$

luego:

$$P(pos|feliz) = \frac{P(feliz|pos)P(pos)}{P(feliz)}$$

Naïve Bayes

Tuits positivos

me encanta la playa

amo la playa

Tuits negativos

odio la playa

mar molesto

token w	frec(pos)	frec(neg)
me	1	0
encanta	1	0
la	2	1
playa	2	1
amo	1	0
odio	0	1
mar	0	1
molesto	0	1
N_{class}	7	5

Naïve Bayes

token w	$P(w\|pos)$	$P(w\|neg)$
me	$\frac{1}{7} \approx 0.14$	0
encanta	$\frac{1}{7} \approx 0.14$	0
la	$\frac{2}{7} \approx 0.28$	$\frac{1}{5} = 0.2$
playa	$\frac{2}{7} \approx 0.28$	$\frac{1}{5} = 0.2$
amo	$\frac{1}{7} \approx 0.14$	0
odio	0	$\frac{1}{5} = 0.2$
mar	0	$\frac{1}{5} = 0.2$
molesto	0	$\frac{1}{5} = 0.2$

Palabras igualmente probables no aportan nada al sentimiento

Naïve Bayes: regularización

token w	$P(w\|pos)$	$P(w\|neg)$
me	0.134	0.01
encanta	0.134	0.01
la	0.274	0.194
playa	0.274	0.194
amo	0.134	0.01
odio	0.01	0.194
mar	0.01	0.194
molesto	0.01	0.194

Naïve Bayes: radio de probabilidades

token w	$P(w\|pos)$	$P(w\|neg)$	ratio(w)
me	0.134	0.01	13.4
encanta	0.134	0.01	13.4
la	0.274	0.194	1.41
playa	0.274	0.194	1.41
amo	0.134	0.01	13.4
odio	0.01	0.194	0.05
mar	0.01	0.194	0.05
molesto	0.01	0.194	0.05

Naïve Bayes: inferencia

Nuevo tuit: $w_1 w_2 w_3 \cdots w_n$

$$\frac{P(pos|w_1 w_2 w_3 \cdots w_n)}{P(neg|w_1 w_2 w_3 \cdots w_n)} = \frac{P(pos)}{P(neg)} \prod_{i=1}^{n} \frac{P(w_i|pos)}{P(w_i|neg)}$$

$$\log \frac{P(pos)}{P(neg)} + \sum_{i=1}^{n} \log \frac{P(w_i|pos)}{P(w_i|neg)}$$

log prior log likehood

evita errores numéricos

Naïve Bayes: sentimiento del tuit

$$\lambda(w) = \log ratio(w), \qquad ratio(w) = \frac{P(w|pos)}{P(w|neg)}$$

Nuevo tuit: $w_1 w_2 w_3 \cdots w_n$

Si las clases son equilibradas, log prior es igual a cero, en este caso:

👍 si:

$$\sum_{i=1}^{n} \lambda(w_i) = log \prod_{i=1}^{n} radio(w_i) > 0$$

👎 si:

$$\sum_{i=1}^{n} \lambda(w_i) = log \prod_{i=1}^{n} radio(w_i) \leq 0$$

Naïve Bayes: entrenamiento

- Recolectar tuits pre-clasificados (conjunto de entrenamiento)
- Pre-procesar los datos
- Establecer el vocabulario de tipos
- Contar las palabras frec(w,class), class $= \{pos, neg\}$
- Para cada palabra y cada clase calcular:

$$P(w|class) = \frac{frec(w, class) + 1}{N_{class} + |V|}$$

- Se calcula $\lambda(w) = log \frac{P(w|pos)}{P(w|neg)}$
- Se calcula log prior

$$\log \frac{P(pos)}{P(neg)} = \log \frac{\text{núm. tuits positivos}}{\text{núm. tuits negativos}}$$

Naïve Bayes: pre-procesamiento

- convertir todo a minúsculas
- remover signos de puntuación, urls, nombres
- remover palabras vacías (*stop words*)
- aplicar *stemming*
- tokenizar las oraciones

Resultado: [aqui, pasa, nada, kino]

Naïve Bayes: clasificador

Nuevo tuit $T \rightarrow [w_1, w_2, \ldots, w_n]$ (pre-procesamiento)

sentimiento(T) = 👍 si:

$$\text{log prior} + \sum_{i=1}^{n} \lambda(w_i) > 0$$

sentimiento(T) = 👎 si:

$$\text{log prior} + \sum_{i=1}^{n} \lambda(w_i) \leq 0$$

Naïve Bayes: supuestos

- Independecia (i.i.d.) ¡falso en PLN!
- Clases relativamente equilibradas