Assignment 0: (vibe checker)

Post A: "Cats are cute and funny."

Post B: "Dogs are funny animals."

Post C: "Cats and Dogs rarely get along."

Corpus: the entire body of the text that we have to analyse.

↓

this corpus is divided into 3 parts.

vocabulary of the corpus: (a list of all unique words)

cats, are, funny, cute, and

dogs, animals

rarely, get, along

in technical terms this bag/list is called a WORD VECTOR.

Note: - ordering is an imp property of vector

- no. of entities in the vector = it's dimension.

Dimension = 10

Word vector = [cats, dogs, rarely, are, animals, get, funny, and, cute, along]

COUNT VECTOR:

a vector created for each post which contains the no. of times a word from the count vector appears in that post.

$A = [1, 0, 0, 1, 0, 0, 1, 1, 1, 0]$

$B = [0, 1, 0, 1, 1, 0, 1, 0, 0, 0]$

$C = [1, 1, 1, 0, 0, 1, 0, 1, 0, 1]$

Stacking these 3 vectors in a matrix:

$$\begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

- Shape = (3 X 10) matrix

Rows = 3 → represents no. of posts.

Columns = 10 → dimension of our vector dataset.

Designing a simple model for classification:

- a probability based model to classify a post into either cat or dog type post.

• Suppose the probability of a word being "cat" is p $P(\text{cat}) = p$

- If post has 10 words

  $P(\text{no cat}) = (1-p)^{10}$

• $P(\text{atleast 1 cat}) = 1 - (1-p)^{10}$

  → this means it is a cat type post.

- If post has L words

  $P(\text{atleast 1 cat}) = 1-(1-p)^{L}$

Calculating the probability vectors:

Total no. of words in corpus = 15

vector ~~de~~ telling how many times each word occurs

[2, 2, 1, 2, 1, 1, 2, 2, 1, 1]

probability vector

$$= \left[\frac{2}{15}, \frac{2}{15}, \frac{1}{15}, \frac{2}{15}, \frac{1}{15}, \frac{1}{15}, \frac{2}{15}, \frac{2}{15}, \frac{1}{15}, \frac{1}{15}\right]$$

from the probability vectors we can say

$$P(\text{cat}) = p = \frac{2}{15}$$

Probability of postA being a cat-type post

$$= 1 - (1-p)^{5}$$

$$= 1 - \left(1 - \frac{2}{15}\right)^{5}$$

$$= 1 - \left(\frac{13}{15}\right)^{5}$$

→ this has nothing to do with the fact that whether postA has the word cat already or not. (i.e Machine knows nothing about that yet)

This just calculates the prob that if we randomly form a 5 word containing sentence from the words in our corpus we have this much prob that it will have atleast one cat in it.

- Now we switch from looking at words & number to looking at the actual documents that is the three posts.

~~How~~ so we need to find the probability that

ⓐ A ~~wor~~ post contains the word "cute" given it is a cat type post.

&

ⓑ Prob of post being cat type when it contains cute.

Let A = Cat-type post

B = Post contains cute-

$P(A) = \frac{2}{3}$ $P(B) = \frac{1}{3}$

$P(A/B) = 1$

$$P(B/A) = \frac{P(A/B) \cdot P(B)}{P(A)}$$

$$= \frac{1 \times \frac{1}{3}}{\frac{2}{3}}$$

$$= \frac{1}{2}$$

- We can say not every cat post mentions cute (because we got 1/2) but every cute post mentions cats (as we got 1).

upvote optimisation :-

Suppose upvote can be approximated as:

$$U(L) = -\frac{L^2}{20} + 3L$$

$$U'(L) = -\frac{L}{10} + 3$$

$\boxed{L=30}$ as it is a downward opening parabola it has only 1 pt where slope $=0$ & ~~wh~~ that will be maxima. So $L=30$ is maximum.

& second derivative test can also be used:

$U''(L) = -1$

$U''(L) < 0$ so max is at $L=30$

Let's define a new function

$$G(L,p) = P(L,p) \cdot U(L)$$

shows expected upvotes from a cat-type post.

finding L,p for maximum G:

p is already know $= \frac{2}{15}$

~~#~~ $P(L,p) = 1-(1-p)^L$

as $L \to \infty$ $P \to 1$ so it's basically constant so $G(L,P) \approx U(L)$

so optimum $L = 30$ only.