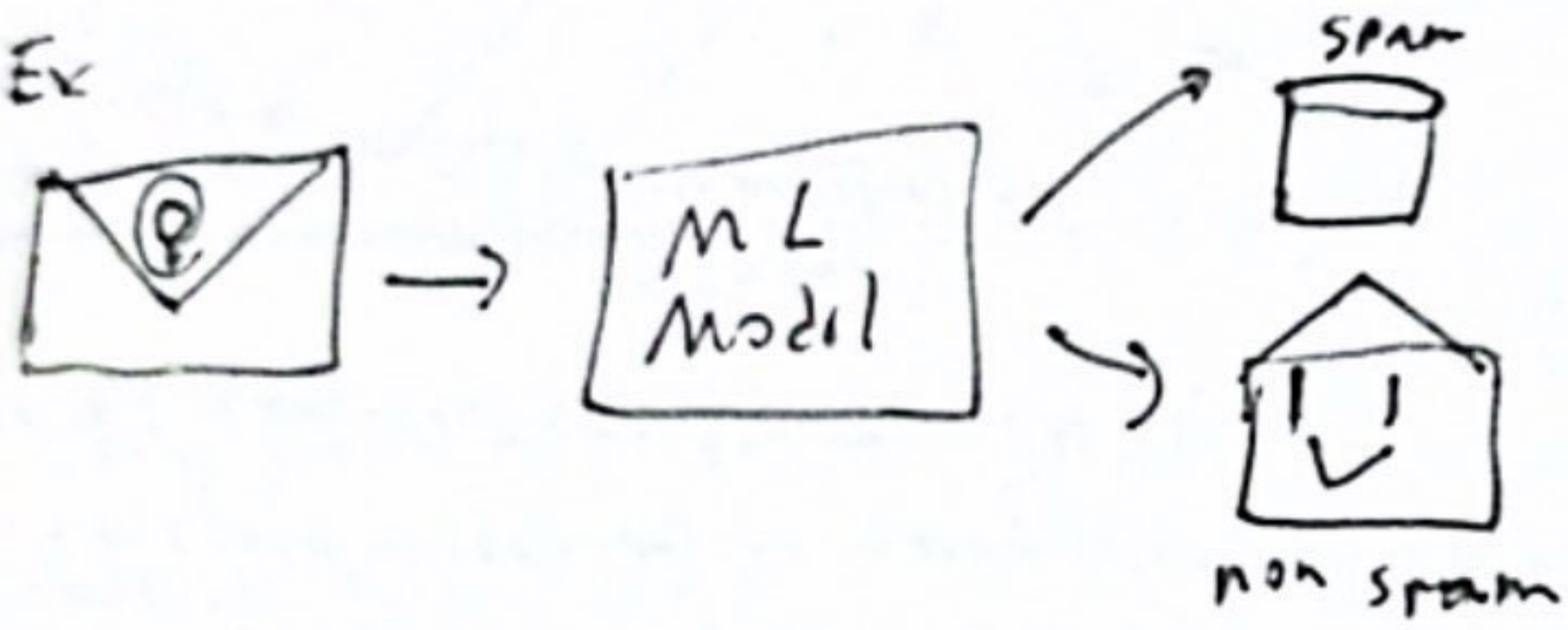
Naive Bayes Classifier (- supervised - classifier)

1 - is a classification algorithm based on bayes theorem it learns probabilities from training to then classify new data based on its probability of falling in some class

2 ✱ Ex spam filters: we can train our algo with many spam and non spam emails and count the occurances of different words in each class (spam / non spam etc) then calculate the probabilities of certian words appering in spam emails and non spam emails. we then classify a new email into spam or non spam based on the words it contains by using bayes theorem

2.1 • Ex



2.2 • train using emails

	label	SMS
0	spam	Party Drugs Prize
1	non spam	Ali how are you
2	spam	Winner Prize!

⋮

↓

	label	Party	Ali	Prize	...
0	spam	1	0	2	...
1	non spam	0	1	0	...
2	spam	0	0	1	...

• count occurrances and find probabilities of words in spam vs non spam

2.3 • Bayes theorem

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

⇓

$P(A|B)$: P of A Given B occured

$P(B|A)$: P of B Given A occured

$P(A)$: P of A occuring

$P(B)$: P of B occuring

2.4 • New email

$P(\text{spam} \mid \text{Party}, \text{Drugs})$ → probability of email being spam based on words "party" and "Drugs"

$$= \frac{P(\text{Party}|\text{spam}) \cdot P(\text{Drugs}|\text{spam}) \cdot P(\text{spam})}{P(\text{party}) \cdot P(\text{Drugs})}$$

2.5 ⇒ Calculate. Like $P(\text{spam}) = 66\% \ldots = X\%$ probability that this email is spam ex 99%

3 ✱ This algo. makes the false assumption that the probabilities of different words appering are indipendent (Feature independence) so its "Naive". so its good for text based classifications "computationally efficient"

$$P(\text{Party}|\text{spam}) = P(\text{Party}|\text{spam} \,\&\, \text{Drugs})$$