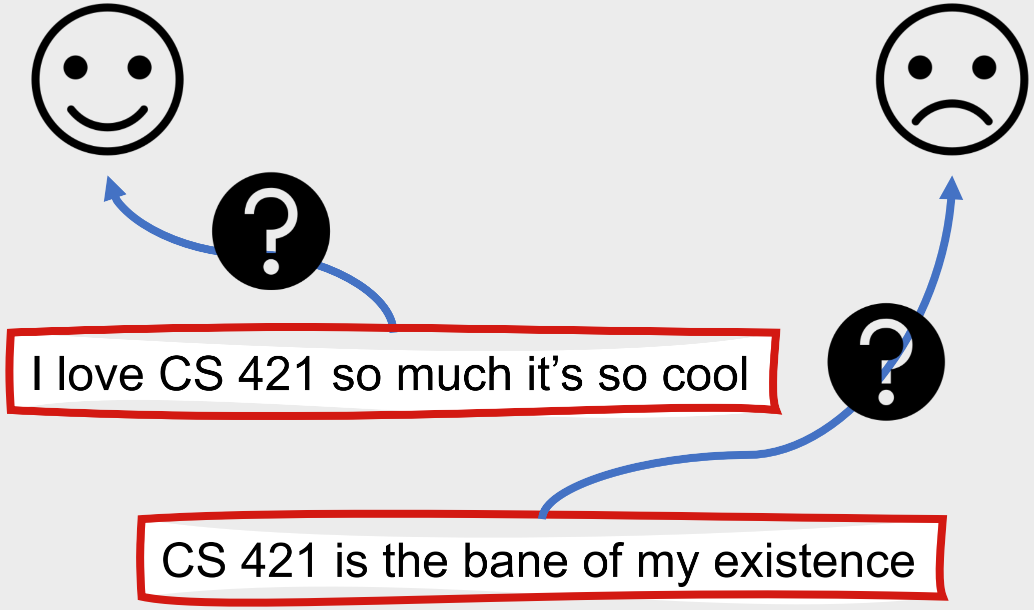
Introduction to Naïve Bayes

Natalie Parde

UIC CS 421

What is Naïve Bayes?

- A **probabilistic classifier** that learns to **predict labels** for new documents



Naïve Bayes Classifiers

Gaussian Naïve Bayes: Assumes the outcomes for the input data are normally distributed along a continuum

Multinomial Naïve Bayes: Assumes the outcomes for the input data follow a multinomial distribution (there is a discrete set of possible outcomes)

Binomial Naïve Bayes: Assumes the outcomes for the input data follow a binomial distribution (there are two possible outcomes)

Multinomial Naïve Bayes

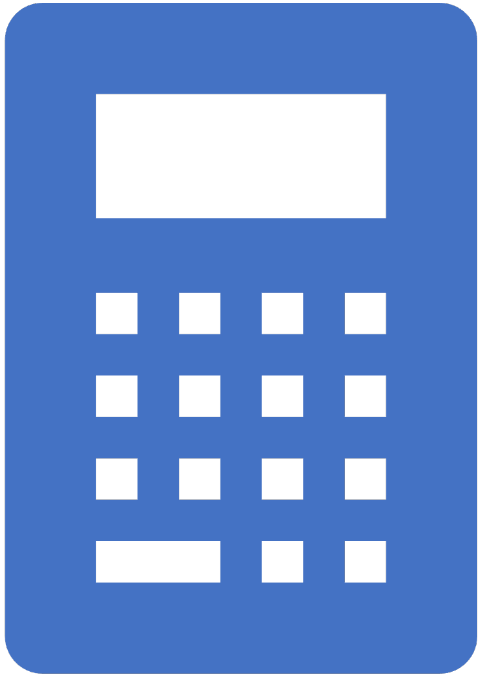
- Each instance falls into one of *n* classes
  - *n*=2 → Binomial Naïve Bayes
- Simple classification based on Bayes' rule
- Simple document representation
  - Technically, any features can be used
  - Traditionally, bag of words features are used

Why is it "Naïve" Bayes?

- Naïve Bayes classifiers make a naïve assumption about how features interact with one another: quite simply, they assume that they don't
- They instead **assume that all features are independent from one another**
- Is this really the case?
  - No---as already seen with language models, words are dependent on their contexts
  - However, Naïve Bayes classifiers still perform reasonably well despite adhering to this naïve assumption

How does it work?

- For a document *d*, out of all classes $c \in C$ the classifier returns the class *c'* which has the maximum **posterior probability**, given the document
  - $c' = \underset{c \in C}{\operatorname{argmax}}\, P(c|d)$



Naïve Bayes computes probabilities using Bayesian inference.

- Bayesian inference uses **Bayes' rule** to transform probabilities like those shown previously into other probabilities that are easier or more convenient to calculate
- Bayes' rule:
  - $P(x|y) = \frac{P(y|x)P(x)}{P(y)}$

Applying Bayesian inference to Naïve Bayes

- If we take Bayes' rule:
  - $P(x|y) = \frac{P(y|x)P(x)}{P(y)}$
- And substitute it into our previous equation:
  - $c' = \underset{c \in C}{\operatorname{argmax}} P(c|d)$
- We get the following:
  - $c' = \underset{c \in C}{\operatorname{argmax}} P(c|d)$

    $= \underset{c \in C}{\operatorname{argmax}} \frac{P(d|c)P(c)}{P(d)}$

How can we simplify this?

- Drop the denominator P(d)
  - We'll be computing $\frac{P(d|c)P(c)}{P(d)}$ for each class, but P(d) doesn't change for each class
    - We're always asking about the most likely class for the same document d
- Thus:
  - $c' = \underset{c \in C}{\operatorname{argmax}} P(c|d)$
    $= \underset{c \in C}{\operatorname{argmax}} P(d|c)P(c)$

What does this mean?

- The most probable class *c'* given some document *d* is the class that has the highest product of two probabilities
  - **Prior probability** of the class P(*c*)
  - **Likelihood** of the document P(*d*|*c*)

$$\boxed{c' = \underset{c \in C}{\operatorname{argmax}} \underbrace{P(d|c)}_{\text{likelihood}} \underbrace{P(c)}_{\text{prior}}}$$

Feature Representation: Intuition

- Represent each document as a **bag of words**
  - Unordered set of words and their frequencies
- Decide how likely it is that a document belongs to a class based on its distribution of **word frequencies**

Usman's poem about Thanksgiving rivaled Natalie's notorious Halloween poem.

's	2
poem	2
Usman	1
about	1
Thanksgiving	1
rivaled	1
Natalie	1
notorious	1
Halloween	1

Bag of Words Features

- Bags of words are sets of features $\{f_1, f_2, \ldots, f_n\}$, where each feature f corresponds to the frequency of one of the words in the vocabulary
- This means that:
  - $c' = \underset{c \in C}{\operatorname{argmax}} P(d|c)P(c) = \underset{c \in C}{\operatorname{argmax}} \underbrace{P(f_1, f_2, \ldots, f_n|c)}_{\text{likelihood}} \underbrace{P(c)}_{\text{prior}}$

The Naïve Bayes assumption means that we can "naïvely" multiply our probabilities for each feature together.

- Why?
  - They're assumed to be independent of one another!
- Therefore:
  - $P(f_1, f_2, \dots, f_n|c) = P(f_1|c) * P(f_2|c) * \cdots * P(f_n|c)$

This brings us to our final equation.

$$c' = \underset{c \in C}{\operatorname{argmax}} P(d|c)P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(f_1, f_2, \ldots, f_n|c)\, P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{f \in F} P(f|c)$$

How do we apply our Naïve Bayes classifier to text?

- Extract bag of words features and insert them into the equation
  - $c' = \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{i \in N} P(f_i|c)$
- To avoid underflow (the generation of numbers that are too tiny to be adequately represented) and increase speed, we usually do these computations in log space:
  - $c' = \underset{c \in C}{\operatorname{argmax}} \log P(c) + \sum_{i \in N} \log P(f_i|c)$

Linear Classifiers

- When we perform these computations in log space, we end up predicting a class as a linear function of the input features
  - $c' = \underset{c \in C}{\operatorname{argmax}} \log P(c) + \sum_{i \in T} \log P(w_i|c)$
- Classifiers that use a linear combination of the inputs to make their classification decisions are called **linear classifiers**
  - Naïve Bayes
  - Logistic Regression