# Decision Tree for Text Categorization

School of Information Studies
Syracuse University

# Text Categorization

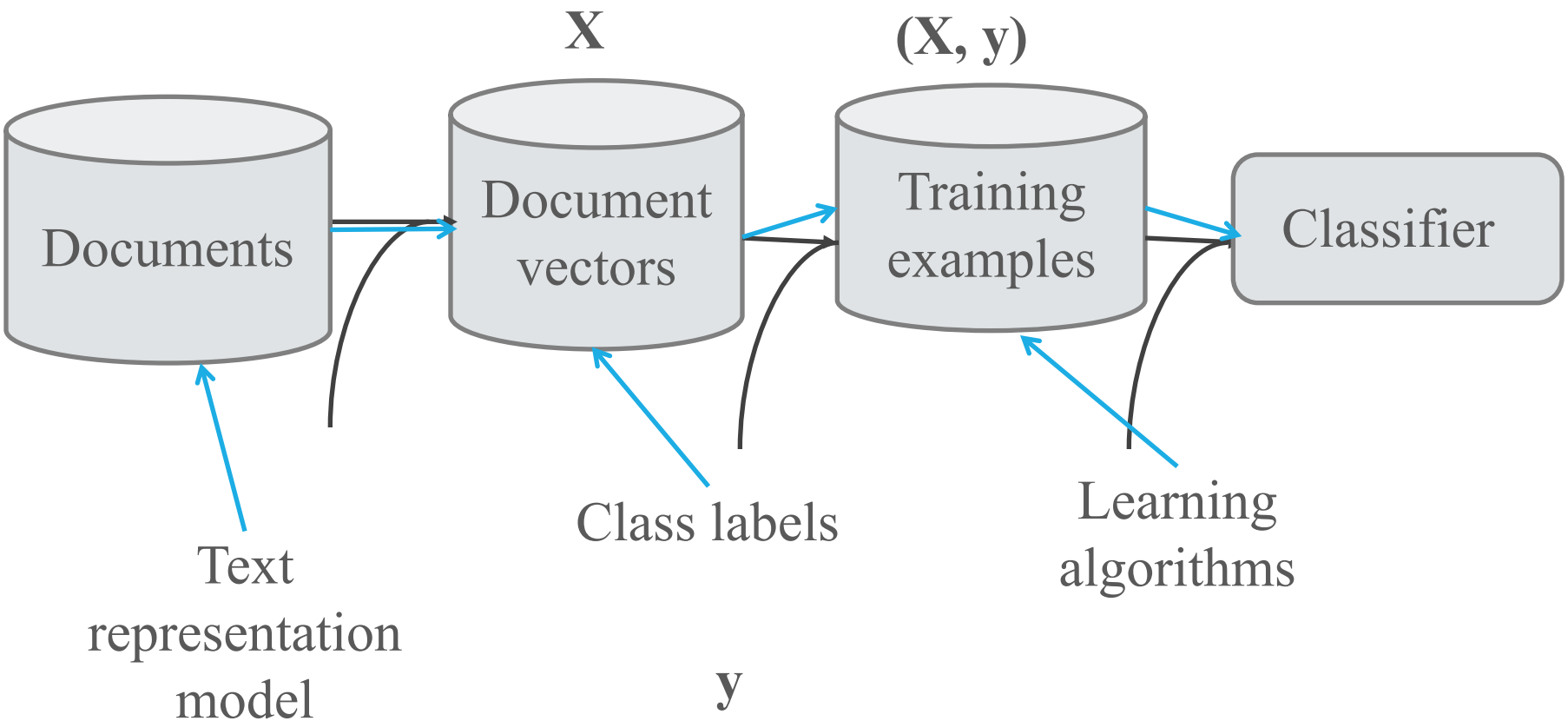
Two steps: training and testing

Step 1: Training

- Goal: build a prediction model (a "classifier") that assigns documents to pre-defined categories (e.g., positive, negative, and neutral comments)
- Input: a collection of training documents and a computer algorithm

| | "happy" | "sad" | "mad" | … | Category |
|---|---|---|---|---|---|
| Doc1 | 1 | 0 | 0 | | positive |
| Doc2 | 0.1 | 0.3 | 0.6 | | negative |
| Doc3 | 0.1 | 0.1 | 0.1 | | neutral |

# Training a Text Classifier



# Step 2: Testing

Goal: use the classifier to predict the category of new documents

Input: a trained classification model and a collection of testing documents with unknown category labels

| | "happy" | "sad" | "mad" | … | Category |
|---|---|---|---|---|---|
| Doc1 | 0.8 | 0.1 | 0.1 | | ? |
| Doc2 | 0.5 | 0.3 | 0.2 | | ? |
| Doc3 | 0.2 | 0.4 | 0.4 | | ? |

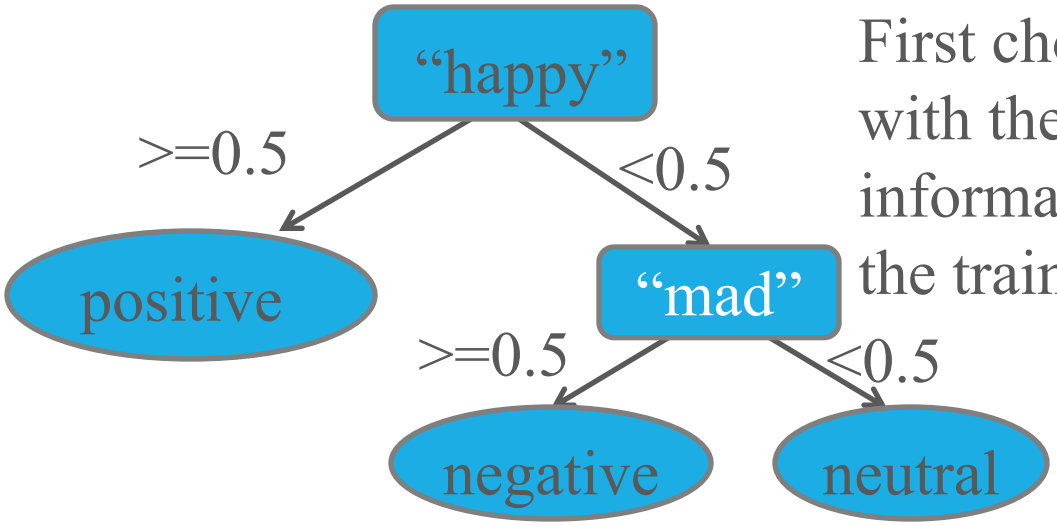
# How to Train a Text "Classifier"?

Many candidate algorithms

- Decision tree
- Naïve Bayes
- Support vector machines
- K-nearest neighbor
- Neural network
- …

# Train Decision Tree Model

| | "happy" | "sad" | "mad" | … | Category |
|---|---|---|---|---|---|
| Doc1 | 1 | 0 | 0 | | positive |
| Doc2 | 0.1 | 0.3 | 0.6 | | negative |
| Doc3 | 0.1 | 0.1 | 0.1 | | neutral |

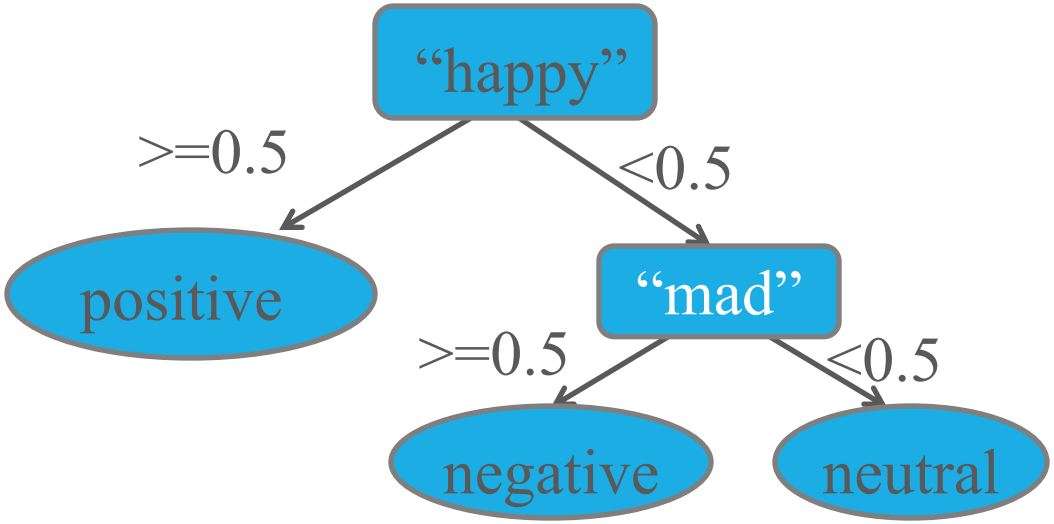



First choose the attribute with the highest information gain to split the training data

# Use Decision Tree Model for Prediction

What is the sentiment of this text document?

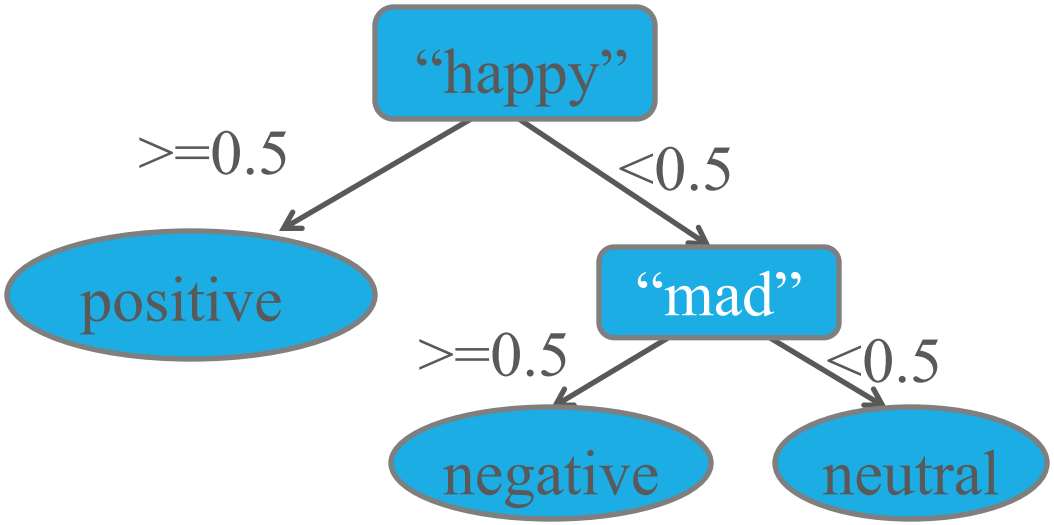
- "*happy happy happy happy mad mad mad sad sad sad*"
- {"happy" = 0.4, "mad" = 0.3, "sad" =0.3}



# Decision Tree Is Not Commonly Used in Text Categorization

A few problems

- Black and white decision: mixed sentiment?
- The tree may become very big: too many word features!



# Multinomial Naïve Bayes for Text Categorization

School of Information Studies
Syracuse University

# Naïve Bayes

Many Naïve Bayes (NB) models

Two common NB models for text classification

- Multinomial model (use word frequency)
- Benoulli model (use word presence/absence)

# Multinomial Naïve Bayes

Pseudo code for MNB in the book *Machine Learning* by Tom Mitchell

# Multinomial Naïve Bayes

LEARN_NAIVE_BAYES_TEXT(*Examples*, $V$)

*Examples is a set of text documents along with their target values. $V$ is the set of all possible target values. This function learns the probability terms $P(w_k|v_j)$, describing the probability that a randomly drawn word from a document in class $v_j$ will be the English word $w_k$. It also learns the class prior probabilities $P(v_j)$.*

*1. collect all words, punctuation, and other tokens that occur in Examples*

- *Vocabulary* ← the set of all distinct words and other tokens occurring in any text document from *Examples*

*2. calculate the required $P(v_j)$ and $P(w_k|v_j)$ probability terms*

- For each target value $v_j$ in $V$ do
  - $docs_j$ ← the subset of documents from *Examples* for which the target value is $v_j$
  - $P(v_j) \leftarrow \frac{|docs_j|}{|Examples|}$
  - $Text_j$ ← a single document created by concatenating all members of $docs_j$
  - $n$ ← total number of distinct word positions in $Text_j$
  - for each word $w_k$ in *Vocabulary*
    - $n_k$ ← number of times word $w_k$ occurs in $Text_j$
    - $P(w_k|v_j) \leftarrow \frac{n_k+1}{n+|Vocabulary|}$

CLASSIFY_NAIVE_BAYES_TEXT(*Doc*)

*Return the estimated target value for the document Doc. $a_i$ denotes the word found in the ith position within Doc.*

- *positions* ← all word positions in *Doc* that contain tokens found in *Vocabulary*
- Return $v_{NB}$, where

$$v_{NB} = \underset{v_j \in V}{\text{argmax}}\, P(v_j) \prod_{i \in positions} P(a_i|v_j)$$

# Bayesian Rule

$P(X, class)$

$= P(X|class) * P(class)$

$= P(class|X) * P(X)$

Our prediction goal

$P(class|X) = P(X|class) * P(class)/P(X)$

# Prior and Conditional Probabilities

Prior probability: P(class)

Conditional probability: P(X|class)

Both can be estimated from training data

# Posterior Probability

Posterior probability: P(class|X)

Calculated based on prior and conditional probabilities using Bayes rules

P(class|X) = P(X|class) * P(class)/P(X)

Ignore P(X) because we just need to find the highest posterior

# Naïve?

Why is this algorithm called "naïve" Bayes?

Because it assumes the occurrence of each word is independent of the occurrence of other words, which is oftentimes not true in text data.

$P(X|class)$

$= P(w_1|class) \times P(w_2|class) \times \ldots \times P(w_n|class)$

# The Independence Assumption

**Not true** for natural language!

Still works quite well on a number of text classification tasks

- Newsgroup classification (Mitchell 1997)
- Movie review classification (Pang et al., 2002)

Theoretical explanation

- Domingos, P., & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine learning, 29(2–3), 103–130.

# Smoothing for Multinomial NB

$$P\left(w_k \middle| v_j\right) \leftarrow \frac{n_k+1}{n+|Vocabulary|}$$

# What Is Stored in a Trained MNB Model?

A look-up table of probabilities

| | Class = 1 | Class = 2 | Class = … |
|---|---|---|---|
| P(class) | 0.40 | 0.60 | |
| $P(w_1|class)$ | 0.75 | 0.50 | |
| $P(w_2|class)$ | 0.25 | 0.67 | |
| $P(w_3|class)$ | 0.33 | 0.50 | |
| $P(w_4|class)$ | 0.80 | 0.33 | |
| … | | | |
| $P(w_n|class)$ | … | … | |

# Benoulli Naïve Bayes for Text Categorization

School of Information Studies
Syracuse University

# Benoulli Model

Multinomial model estimates the probability of the event that one of the N unique words occurs in a position.

Benoulli model estimates the probability of the event that a word is present or absent.

# Comparison 1: Priors Are the Same

**Table 13.1:** Data for parameter estimation examples.

| | | docID | words in document | in $c$ = China? | |
|---|---|---|---|---|---|
| | training set | 1 | Chinese Beijing Chinese | yes | |
| | | 2 | Chinese Chinese Shanghai | yes | |
| | | 3 | Chinese Macao | yes | |
| | | 4 | Tokyo Japan Chinese | no | |
| | test set | 5 | Chinese Chinese Chinese Tokyo Japan | ? | |

Priors are the same: ¾ and ¼

# Comparison 2: Conditional Probabilities Are Different

In multinomial model, conditional probabilities are based on word frequency, smoothed over the vocabulary.

$$\hat{P}(\text{Chinese}|c) = (5+1)/(8+6) = 6/14 = 3/7$$

In Benoulli model, conditional probabilities are based on document frequency, smoothed over two events, either presence or absence.

$$\hat{P}(\text{Chinese}|c) = (3+1)/(3+2) = 4/5$$

# Comparison 3: Posteriors Are Different

| test set | 5 | Chinese Chinese Chinese Tokyo Japan | ? |
|---|---|---|---|

Posterior in Benoulli model

$$\begin{aligned}\hat{P}(c|d_5) &\propto \hat{P}(c)\cdot\hat{P}(\text{Chinese}|c)\cdot\hat{P}(\text{Japan}|c)\cdot\hat{P}(\text{Tokyo}|c)\\ &\quad\cdot(1-\hat{P}(\text{Beijing}|c))\cdot(1-\hat{P}(\text{Shanghai}|c))\cdot(1-\hat{P}(\text{Macao}|c))\\ &= 3/4\cdot 4/5\cdot 1/5\cdot 1/5\cdot(1-2/5)\cdot(1-2/5)\cdot(1-2/5)\\ &\approx 0.005\end{aligned}$$

Posterior in multinomial model

$$\begin{aligned}\hat{P}(c|d_5) &\propto 3/4\cdot(3/7)^3\cdot 1/14\cdot 1/14\approx 0.0003.\\ \hat{P}(\bar{c}|d_5) &\propto 1/4\cdot(2/9)^3\cdot 2/9\cdot 2/9\approx 0.0001.\end{aligned}$$

# Which NB to Choose?

McCallum, A., & Nigam, K. (1998, July). A comparison of event models for naïve Bayes text classification. In *AAAI-98 workshop on learning for text categorization* (Vol. 752, No. 1, pp. 41–48).

- Benoulli for shorter texts (can take Boolean representation only)
- Multinomial for longer texts (can take word count, tfidf)

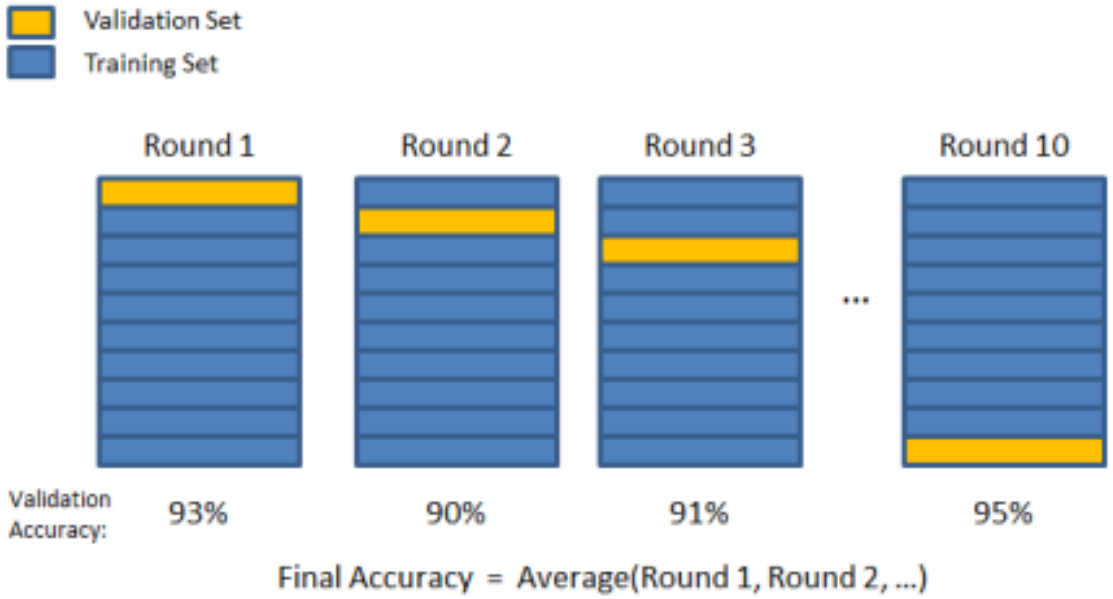
# Classification Model Evaluation

School of Information Studies
Syracuse University

# Evaluation

Hold-out test

Cross validation



# Benchmark Data Sets for Evaluating Text Categorization Algorithms

## Reuters Collection

- News articles with topic categories taken from Reuters newswire
- The topic categories are economic subject categories. Examples include "coconut," "gold," "inventories," and "money-supply."

## 20 News Groups

- Usenet articles taken from 20 news groups
- alt.atheism, comp.graphics, comp.os.ms-windows.misc , comp.sys.ibm.pc.hardware , comp.sys.mac.hardware , comp.windows.x , misc.forsale rec.autos, rec.motorcycles, rec.sport.baseball , rec.sport.hockey , sci.crypt, sci.electronics , sci.med, sci.space , soc.religion.christian, talk.politics.guns, talk.politics.mideast , talk.politics.misc talk.religion.misc

## Cornell Movie Review Data