Lecture 4: Spam Filtering

Siddharth Garg

sg175@nyu.edu

Spam Detection: Features

- Recall features used in the UCI Spam database

  48 continuous real [0,100] attributes of type word_freq_WORD

- Even easier way to encode features:
  - x_i = 1 if term *i* appears in a document; 0 otherwise
  - Boolean features

- Assume M Boolean features, x = (x_1, x_2,…, x_M)
  - We want to map this M-dimensional Boolean input to a Boolean output *y*
  - *Thoughts?*
  - Instead of using LR (or SVM) we will start with an even simpler approach referred to as "Naïve Bayes"

Ref. Metsis, Vangelis, Ion Androutsopoulos, and Georgios Paliouras. "Spam filtering with naive bayes-which naive bayes?." In *CEAS*, vol. 17, pp. 28-69. 2006.

Naiive Bayes for Spam Filtering

- Assume M Boolean feature, x = (x_1,x_2,…, x_M)
- Each email is either {s=spam,l=legit}
- We begin by computing:

"Bernoulli Naiive Bayes"

$$P\{spam \mid x\} = \frac{P\{x \mid spam\} * P\{spam\}}{P\{x\}}$$

Bayes Rule

$$\boxed{P\{A \cap B\} = P\{A \mid B\} * P\{B\}}$$

Ref: Metsis, Vangelis, Ion Androutsopoulos, and Georgios Paliouras. "Spam filtering with naive bayes-which naive bayes?." In *CEAS*, vol. 17, pp. 28-69. 2006.

Naiive Bayes for Spam Filtering

- We begin by computing:

$$P\{spam \mid x\} = \frac{P\{x \mid spam\} * P\{spam\}}{P\{x\}}$$

$$P\{x_1, x_2, .., x_M \mid spam\} = P\{x_1 \mid spam\} * P\{x_2 \mid spam\} * .. * P\{x_M \mid spam\}$$

Assuming that term occurrences are independent (given class

Is this a reasonable assumption?

Naiive Bayes for Spam Filtering

$$P\{x_1 \mid spam\} * P\{x_2 \mid spam\} * .. * P\{x_M \mid spam\}$$

How do we estimate this from the training dataset?

$$P\{x_1 = 1 \mid spam\} = p_{i,s}$$

=(#Spam emails that contain term i)/(#spam emai

What happens if term i never
occurred in any spam email in the

Laplacian Smoothing

$p_{i,s}$ ~~(#Spam emails that contain term 1)/(#spam emails)~~

= (#Spam emails that contain term i+1)/(#spam emails+2)

Equivalent to assuming two addition spam emails in the training dataset, of which on contains all terms and the other is empty

$$P\{x_1 = 0 \mid spam\} = 1 - p_{i,s}$$

$$P\{x_1, x_2, .., x_M \mid spam\} = \prod_{i=1}^{M} p_{i,s}^{x_i} (1 - p_{i,s})^{1-x_i}$$

Naiive Bayes for Spam Filtering

$$P\{x\} = P\{spam\} * P\{x \mid spam\} + P\{legit\} * P\{x \mid legit\}$$

$$P\{spam \mid x\} = \frac{P\{x \mid spam\} * P\{spam\}}{P\{x\}}$$ Vs. $$P\{legit \mid x\} = \frac{P\{x \mid legit\} * P\{legit\}}{P\{x\}}$$

Or:

$$P\{spam \mid x\} \geq threshold$$

Spam Detection: Occurences

- Recall features used in the UCI Spam database

  48 continuous real [0,100] attributes of type word_freq_WORD

- Let’s consider a different representation that is closer to the UCI spambase features: Term Frequencies (TF)
  - x_i # times term *i* appears in a document ($x_i \in \aleph$)
  - Each document is represented by x = (x_1, x_2, …, x_M), a vector of term frequencies
  - We will again use a Naïve Bayes approach to classify documents as either spam or legit
    - “**Multinomial Naïve Bayes**”

Applying Bayes Rule

$$P\{spam \mid x\} = \frac{P\{x \mid spam\} * P\{spam\}}{P\{x\}}$$

$$P\{x_1 \mid spam\} * P\{x_2 \mid spam\} * .. * P\{x_M \mid spam\}$$

Independence assumption shows up again!

But how do we estimate the probability: $P\{x_1 = t \mid spam\}$

What if there is no document in the training dataset where term 1 occurs *t* time

“Bag of Words” Model

Bag containing *M* terms

Term *i* picked with probability $p_{i,s}$

$$\sum_{i=1}^{M} p_{i,s} = 1$$

Term i picked with probability $p_{i,s}$

Term i picked with probability $p_{i,s}$

1

2

D

Spam Email Containing D terms/words

Likelihood Estimation

- Say you have a spam e-mail of length D generated using the BoW model

Bag containing *M* terms

Term i picked with probability $p_{i,s}$

$\sum_{i=1}^{M} p_{i,s} = 1$

Term i picked with probability $p_{i,s}$

Term i picked with probability $p_{i,s}$

1 2 D

Spam Email Containing D terms/words

Feature vector x

$$P\{x \mid spam, D\} = \prod_{i=1}^{M} (p_{i,s})^{x_i}$$

Assuming term and positional independence

Are we done?

Likelihood Estimation

- Recall that the BoW model does not keep track of the positions in which terms appear
  - We must account for all possible ways of arranging
    - x_1 instances of term 1 and
    - x_2 instances of term 2 and
    - … x_M instances of term M into D locations

$$
{}^{D}C_{x_1} * {}^{D-x_1}C_{x_2} * \ldots * {}^{D-x_1-x_2-\ldots x_{M-1}}C_{x_M}
$$

Choose x_1 locations from a total of D locations

Choose x_2 locations from remaining D- x_1 locations

Likelihood Estimation

$$ {}^{D}C_{x_1} * {}^{D-x_1}C_{x_2} * .. * {}^{D-x_1-x_2-..x_{M-1}}C_{x_M} = \frac{D!}{x_1!(\cancel{D-x_1})!} * \frac{(\cancel{D-x_1})!}{x_2!(\cancel{D-x_1}-x_2)!} ...1 $$

$$ = \frac{D!}{x_1!x_2!..x_M!} $$

$$ P\{x \mid spam, D\} = D! \prod_{i=1}^{M} \frac{(p_{i,s})^{x_i}}{x_1!} $$

[Typo: this should be x_{i}]

Note that this expression is conditioned on the length of the e-mail *D*. In practice, emails can be of varying lengths.

Accounting for Document Length

$$P\{x \mid spam\} = P\{x \mid spam, D\} P\{D \mid spam\} = \boxed{P\{x \mid spam, D\} P\{D\}}$$

Assume email length is independent of whether email is spam or legit.

Putting it all together:

$$P\{spam \mid x\} = \frac{P\{x \mid spam, D\} \cancel{P\{D\}} P\{spam\}}{\cancel{P\{x\}}}$$ **Vs.** $$P\{legit \mid x\} = \frac{P\{x \mid legit, D\} \cancel{P\{D\}} P\{legit\}}{\cancel{P\{x\}}}$$

Estimating Model Parameters

Note: this is **not** the probability of term *i* appearing in a spam document

Bag containing *M* terms

Term i picked with probability $p_{i,s}$

$$\sum_{i=1}^{M} p_{i,s} = 1$$

Term i picked with probability $p_{i,s}$

Term i picked with probability $p_{i,s}$

Probability that a randomly selected word in a spam email is term *i*

1	2		D

$$p_{i,s} = \frac{\#\text{ occurrences of term } i \text{ in spam}}{\sum_{i=1}^{M} \#\text{ occurrences of term } i \text{ in spam}}$$

Spam Email Containing D terms/words

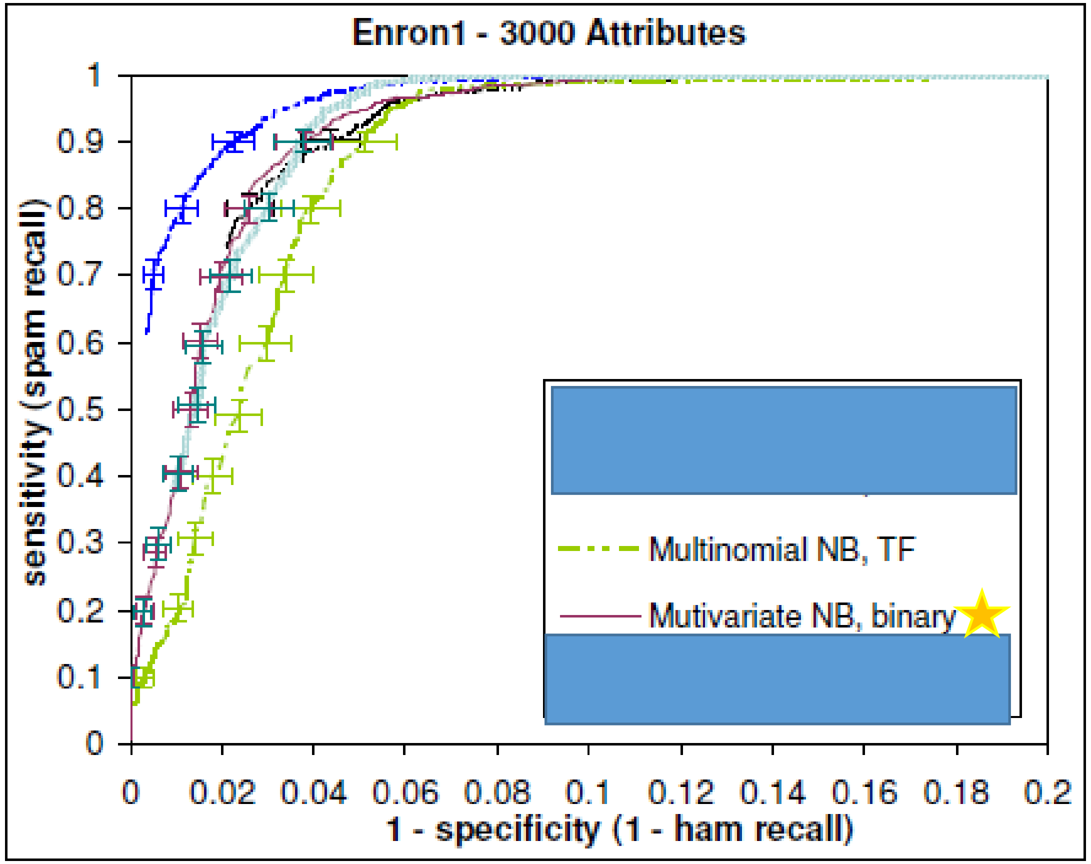
$$p_{i,s} = \frac{1 + \#\text{ occurrences of term } i \text{ in spam}}{M + \sum_{i=1}^{M} \#\text{ occurrences of term } i \text{ in spam}}$$

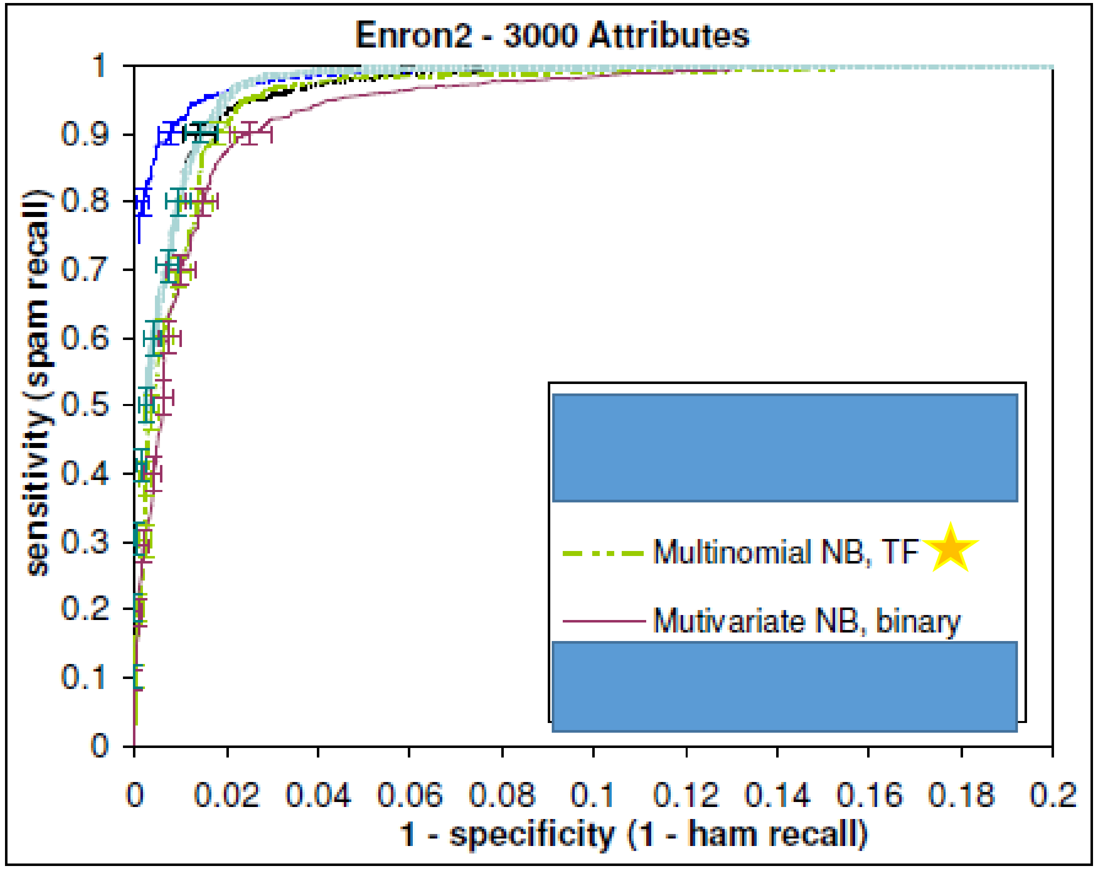
With Laplacian smoothing

Bernoulli NB Vs. Multinomial NB with TF

- Data for 6 different users from ENRON dataset
  - Augmented with spam emails from various sources (legit = “ham”)
  - Top-3000 features selected (we will discuss feature selection soon)

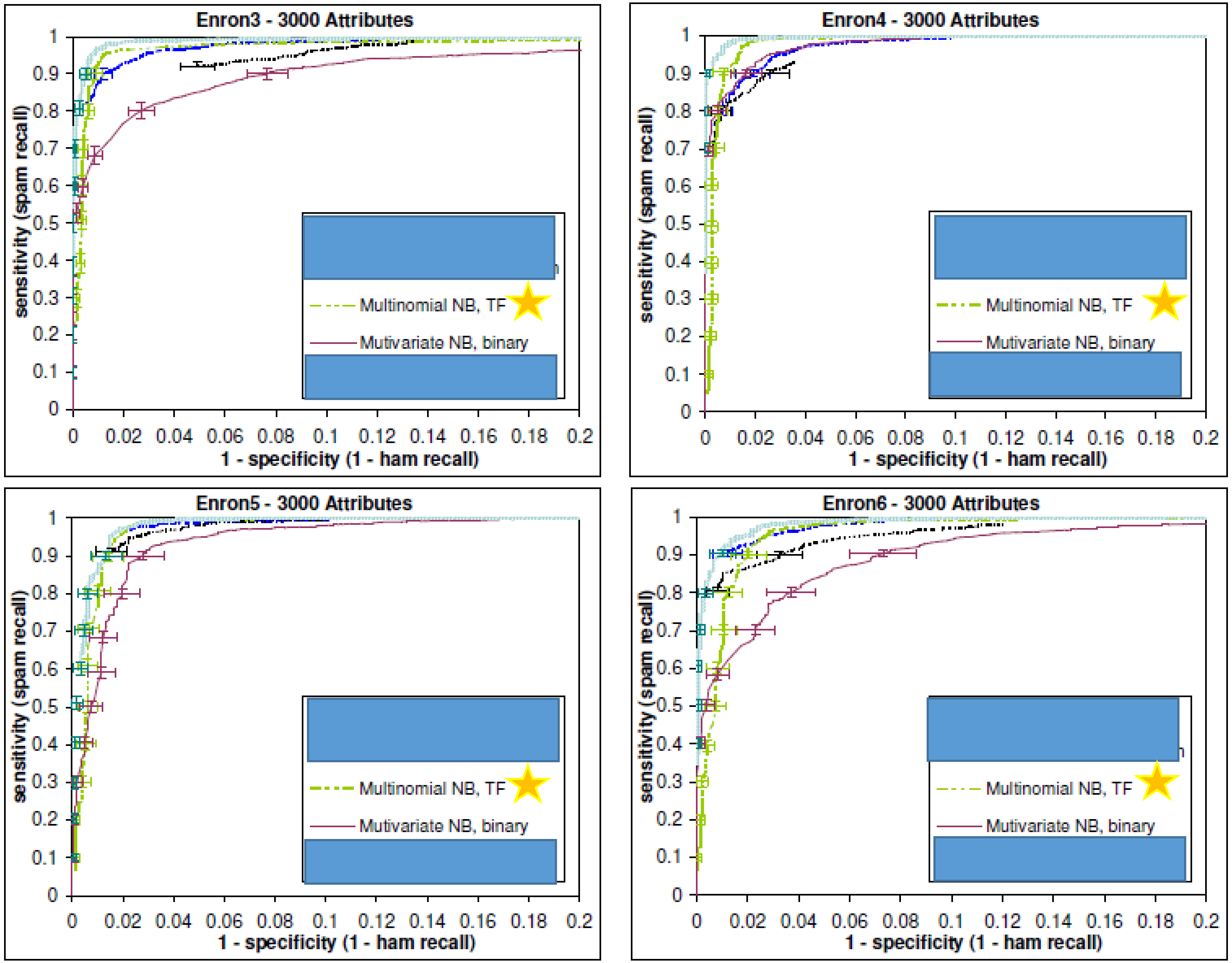
% of spam emails predicted as spam





% of legit emails classified as spam

Bernoulli NB Vs. Multinomial NB with TF



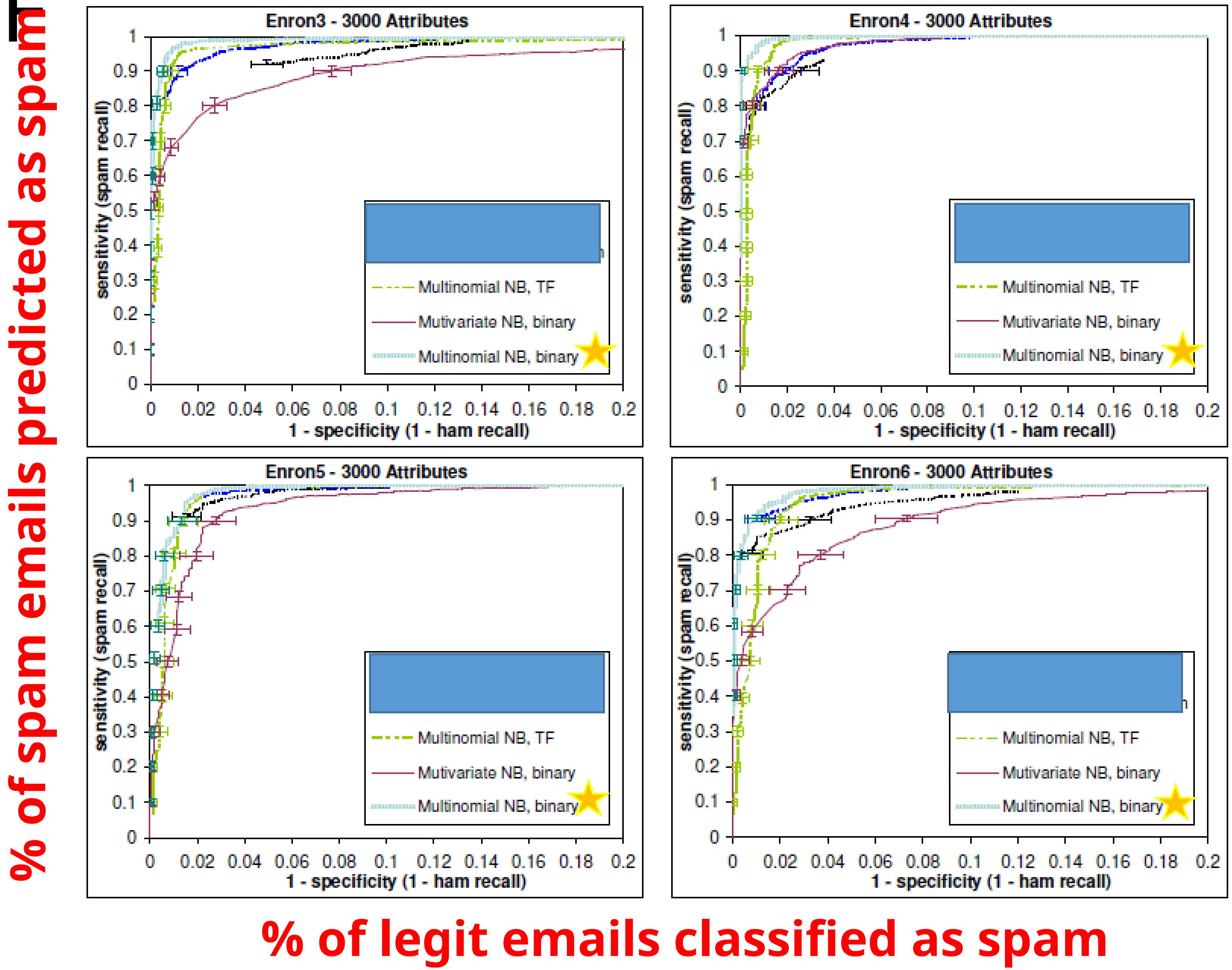
% of spam emails predicted as spam

% of legit emails classified as spam

Tempted to conclude that using term frequencies instead of binary occurrences helped in spam filtering.

Is there any other reason multinomial NB with TF might have outperformed Bernoulli NB?

Bernoulli NB Vs. Multinomial NB with TF



Multinomial NB with Binary instead of TF features performs the best!

Multinomial NB with Binary Features

- Let x = (x_1, x_2, ..., x_M) be the TF features. Binary features are derived from the TF features as follows:

$$\overline{x} = (\overline{x_1} = \min(1, x_1), \overline{x_2} = \min(1, x_2), \ldots, \overline{x_M} = \min(1, x_M))$$

- Transformation is applied to both the training and test data and the multinomial model is used for prediction, i.e.,

$$P\{\overline{x} \mid spam\} = p(D)D! \prod_{i=1}^{M} \frac{(p_{i,s})^{\overline{x_i}}}{\overline{x_1}!} \quad \left\{ \begin{array}{lll} p_{i,s} & \text{if} & \overline{x_i} = 1 \\ 1 & \text{if} & \overline{x_i} = 0 \end{array} \right.$$

[Typo: this should be x_{i}]

Multinomial Vs. Bernoulli NB

Multinomial

$$P\{\bar{x} \mid spam\} = \cancel{p(D)} D! \prod_{i=1}^{M} (p_{i,s})^{\bar{x}_i}$$

Bernoulli

$$P\{x \mid spam\} = \prod_{i=1}^{M} p_{i,s}^{x_i} \, (1 - p_{i,s})^{1 - x_i}$$

How are the two different?

1. Multinomial model <u>*ignores negative evidence*</u>
2. $p_{i,s}$ is estimated differently

$$\frac{1 + \#\text{ occurrences of term } i \text{ in spam}}{M + \sum_{i=1}^{M} \#\text{ occurrences of term } i \text{ in spam}}$$

$$\frac{1 + \#\text{Spam emails that contain term } i}{2 + \#\text{spam emails}}$$

Why Ignore Negative Evidence?

Schneider, Karl-Michael. "On word frequency information and negative evidence in Naive Bayes text classification." *Advances in Natural Language Processing*. Springer, Berlin, Heidelberg, 2004. 474-485.

Table 1. Statistics of the ling-spam corpus

	Total	Ling	Spam
Documents	2893	2412 (83.4%)	481 (16.6%)
Vocabulary	59,829	54,860 (91.7%)	11,250 (18.8%)

Vocabulary	Total		Ling		Spam	
	Words	Documents	Words	Documents	Words	Documents
Full	226.5	11.0	226.9	9.1	224.5	1.8
MI 5000	138.5	80.2	133.8	64.5	162.5	15.6
MI 500	44.0	254.5	39.6	190.9	66.2	63.7

Observation 1: >80% of words never occur in spam documents, while only 10% of words never occur in legit documents

Observation 2: On average, documents only contain a very small fraction of words from the vocabulary

For Bernoulli NB, probability of a document is mostly determined by words that do not appear in the document!

Why is Multinomial Binary Features better than Term Frequencies?

Multinomial TF assumes repeated instances of the same word occur independently, but that is not the case -> for example, if a word appears once it is more likely to appear multiple times. Therefore multinomial TF is a poor model for the underlying data.