A lecture on **Linguistics and Artificial Intelligence**
in the series, "Linguistics: An interdisciplinary field"

Context:
Meaning is the next frontier in computational linguistics. By far the most fertile approaches to detailed meaning analysis have been structure-based, posing the problem of meaning analysis as the construction of a structured formula or a graph. The construction of such logical representations has in turn often depended on inferring syntactic trees or syntactic graphs that directly represent the grammatical relations of parts of a sentence, but recent research has begun to question whether this intermediary syntactic step is really necessary. Professor Noah Smith's group has been working on the general problem of applying deep learning methods to structure prediction.

Linguistic Structure Prediction and Representation Learning

Professor Noah Smith

Paul G. Allen College of Computer Science
and Engineering, University of Washington

Abstract:
Linguistic structure prediction infers abstract representations of text, like syntax trees and semantic graphs, enabling interpretation in applications like question answering, information extraction, and opinion analysis. This talk is about the latest family of methods for linguistic structure prediction, which make heavy use of representation learning via neural networks. I'll present these new methods as continuous generalizations of state machines and probabilistic grammars. I'll show how they've led to fast and accurate performance on several syntactic and semantic parsing problems.

March 2, 2018, Lecture: 2 to 3 PM SH 113
Meeting with students: 3 to 4 PM SHW 237

Sponsored by the Department of Linguistics and Asian/Middle Eastern Languages, College of Arts and Letters, and the Linguistics Student Association

Homework

- Read chapters 1, 2, 5 6, 7 in *Introduction to Machine Learning with Python*

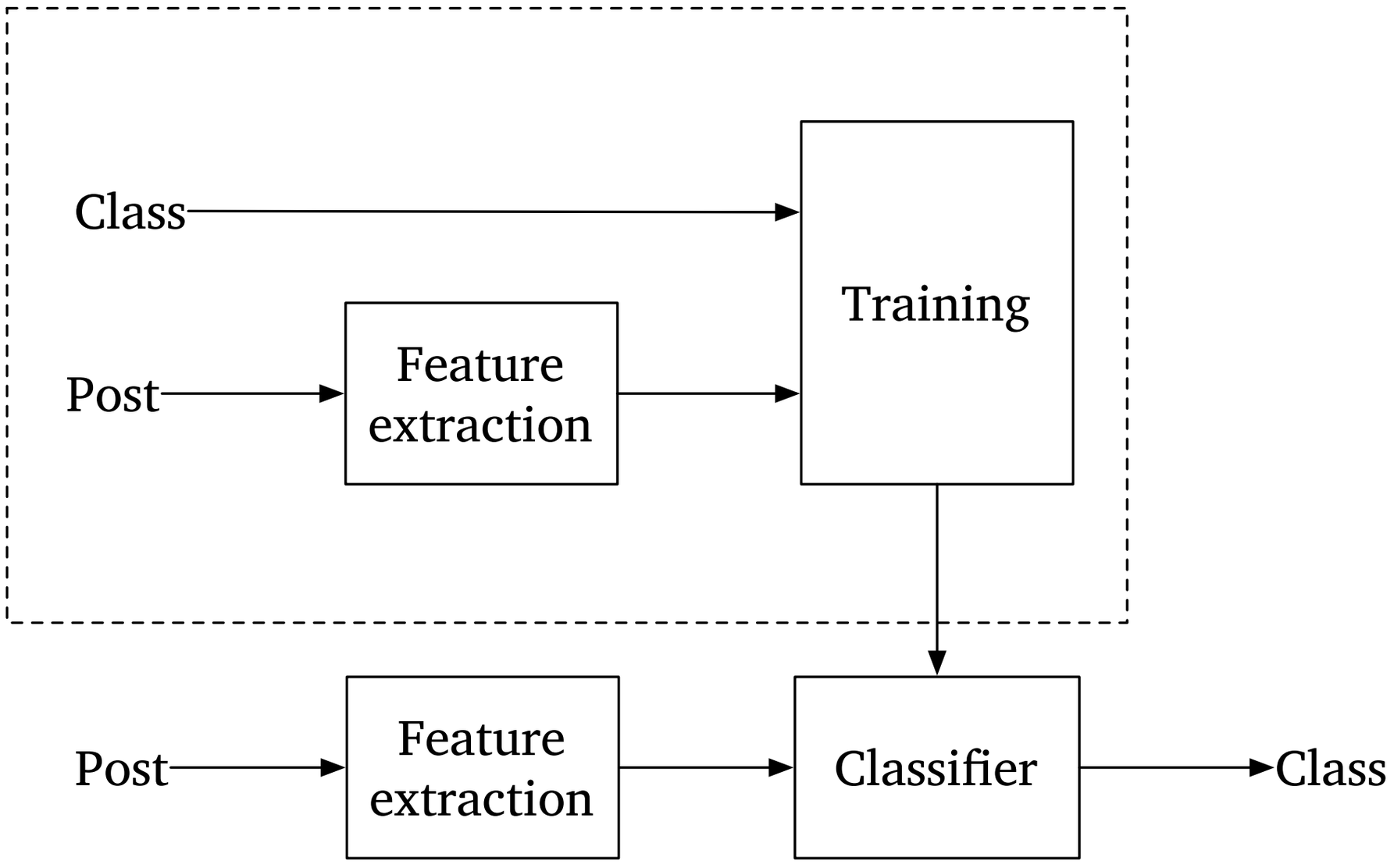
Text classification

- Text classification, using '20 Newsgroups' dataset
- 20,000 postings selected from 20 Usenet groups:

```
comp.graphics                    sci.electronics
comp.os.ms-windows.misc          sci.med
comp.sys.ibm.pc.hardware         sci.space
comp.sys.mac.hardware            misc.forsale
comp.windows.x                   talk.politics.misc
rec.autos                        talk.politics.guns
rec.motorcycles                  talk.politics.mideast
rec.sport.baseball               talk.religion.misc
rec.sport.hockey                 alt.atheism
sci.crypt                        soc.religion.christian
```

- Can we write a program that guesses which newsgroup a post comes from?

Text classification



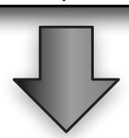
Text classification

- To build a classifier:
  - Extract **features** from items to be classified
    - Binary lexical features
  - **Select** a subset of potential features
    - Use all available features
  - Choose a model class and **train**
    - Naive Bayes

Text classification

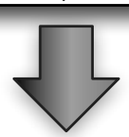
```
Say, you bought your Saturn at $13k, with a dealer profit of $2k.
If the dealer profit is $1000, then you would only be paying $12k for
the same car.  So isn't that saving money?

Moreover, if Saturn really does reduce the dealer profit margin by $1000,
then their cars will be even better deals.  Say, if the price of a Saturn was
already $1000 below market average for the class of cars, then after they
reduce the dealer profit, it would be $2000 below market average.  It will:
```



```
say , you bought your saturn at $13k , with a dealer profit of $2k .
if the dealer profit is $1000 , then you would only be paying $12k for
the same car .  so isn 't that saving money ?

moreover , if saturn really does reduce the dealer profit margin by $1000 ,
then their cars will be even better deals .  say , if the price of a saturn was
already $1000 below market average for the class of cars , then after they
reduce the dealer profit , it would be $2000 below market average .  it will :
```



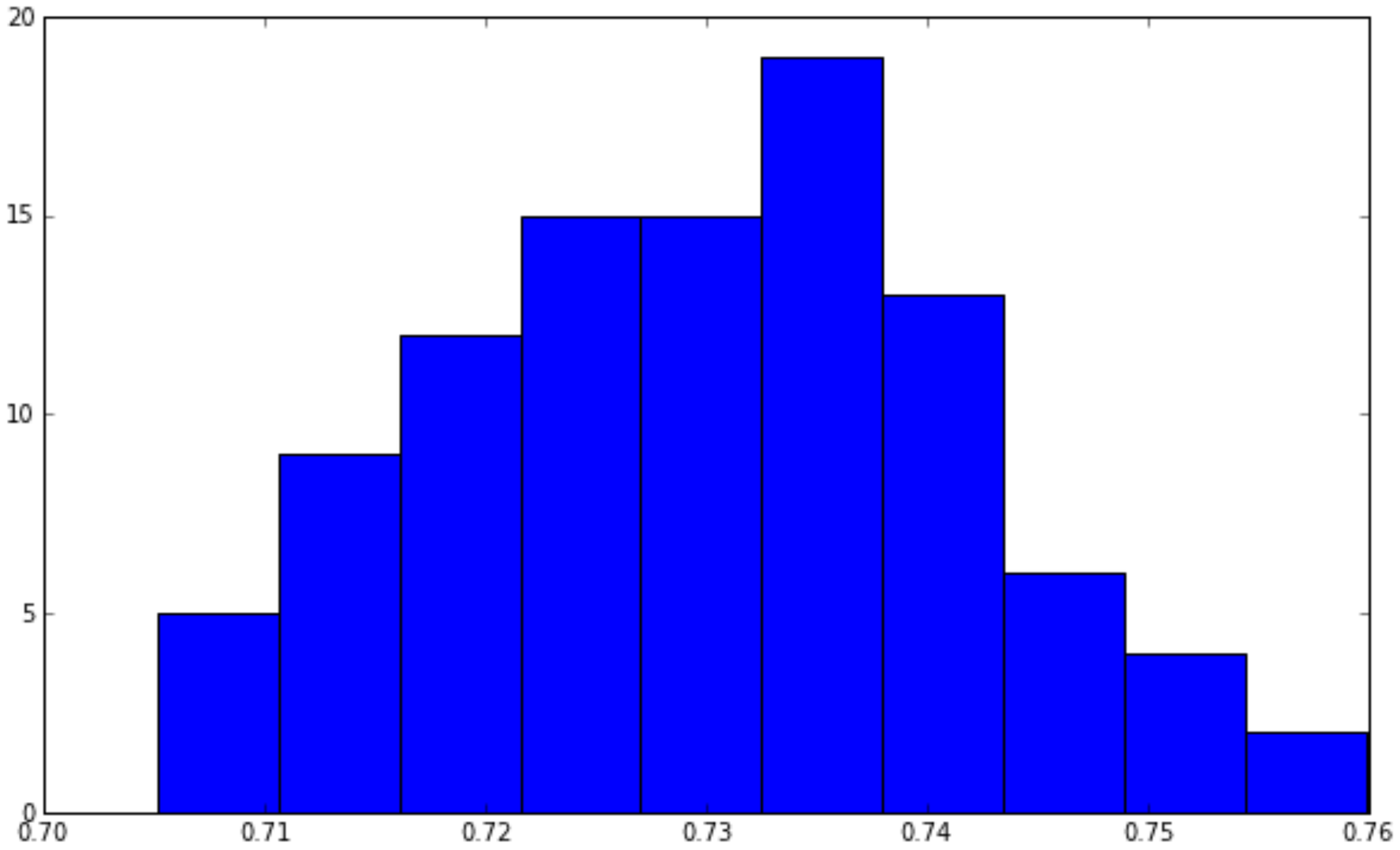
```
a after already at average be below
better bought by car cars class dealer
deals does even for if is isn it
margin market money moreover of only
paying price profit really reduce same
saturn saving say so that the their
then they was will with would you your
```

Evaluation

- Train on 15,064 posts and test on 1,882 posts
- We use accuracy on the evaluation set as an estimate for accuracy
- Baseline (guess most common class) yields 4.09%
- Overall: 74.07% accuracy
- Is this good?

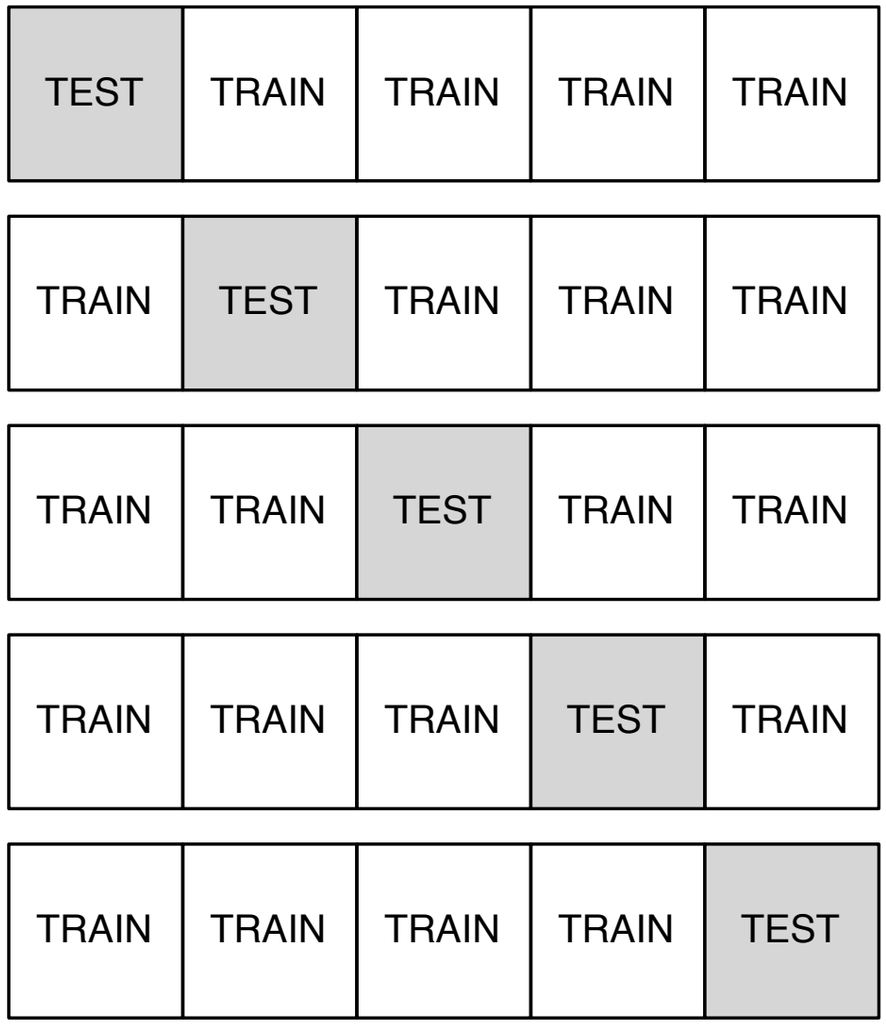
Evaluation

- The score we get depends on how we split the data
- 100 runs with different training / eval splits yield avg accuracy of 72.97%



Evaluation

- To reduce randomness, we can use ***k*-fold cross validation**



- For some methods, **leave one out** cross-validation is practical

Evaluation

- We can judge the performance of a classifier by its **accuracy** (the fraction of instances which it correctly labels) or error (1–accuracy)
- For each class, we count false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN)

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

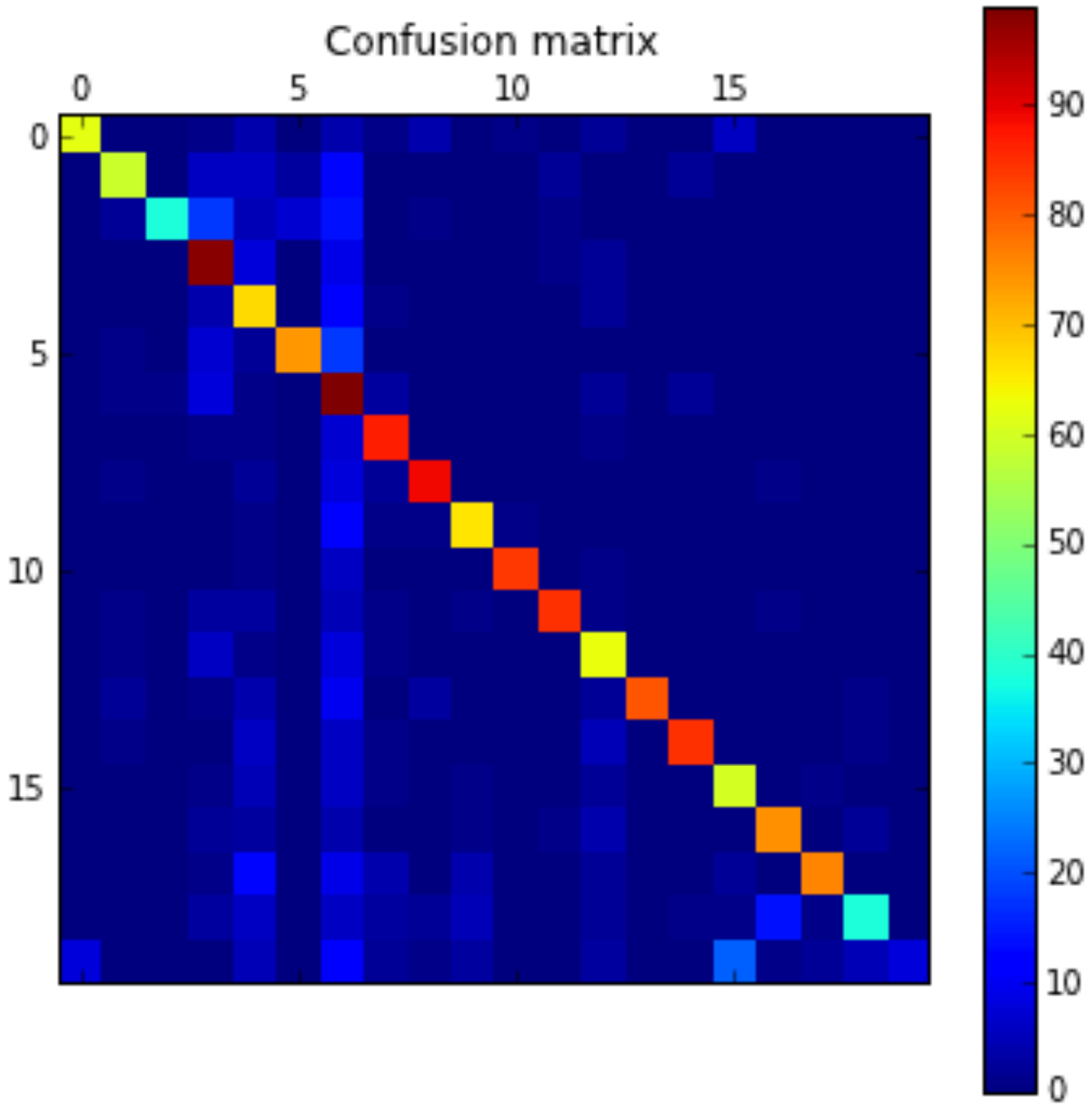
$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F}_1\text{-score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Error analysis

	precision	recall	f-score	support
alt.atheism	0.89	0.73	0.80	85
comp.graphics	0.86	0.65	0.74	91
comp.os.ms-windows.misc	0.97	0.44	0.61	86
comp.sys.ibm.pc.hardware	0.61	0.83	0.71	118
comp.sys.mac.hardware	0.47	0.78	0.58	86
comp.windows.x	0.88	0.73	0.80	102
misc.forsale	0.37	0.85	0.52	117
rec.autos	0.81	0.90	0.85	97
rec.motorcycles	0.88	0.86	0.87	103
rec.sport.baseball	0.81	0.80	0.81	82
rec.sport.hockey	0.98	0.91	0.94	92
sci.crypt	0.94	0.84	0.89	101
sci.electronics	0.67	0.79	0.72	80
sci.med	1.00	0.78	0.88	104
sci.space	0.94	0.81	0.87	105
soc.religion.christian	0.66	0.78	0.71	77
talk.politics.guns	0.82	0.82	0.82	92
talk.politics.mideast	0.95	0.68	0.80	111
talk.politics.misc	0.81	0.46	0.59	82
talk.religion.misc	1.00	0.11	0.20	71
avg / total	0.81	0.74	0.74	1882

Error analysis



Feature selection

- Feature selection prunes the feature space to give the learning algorithm an easier task
- Most frequent lexical features:

from	15,064
subject	15,064
the	13,992
a	13,172
to	13,143
in	12,679
i	12,576
and	12,477
of	12,419
is	11,470

- Total of 93,122 features, but 42,086 features occur only once

Feature selection

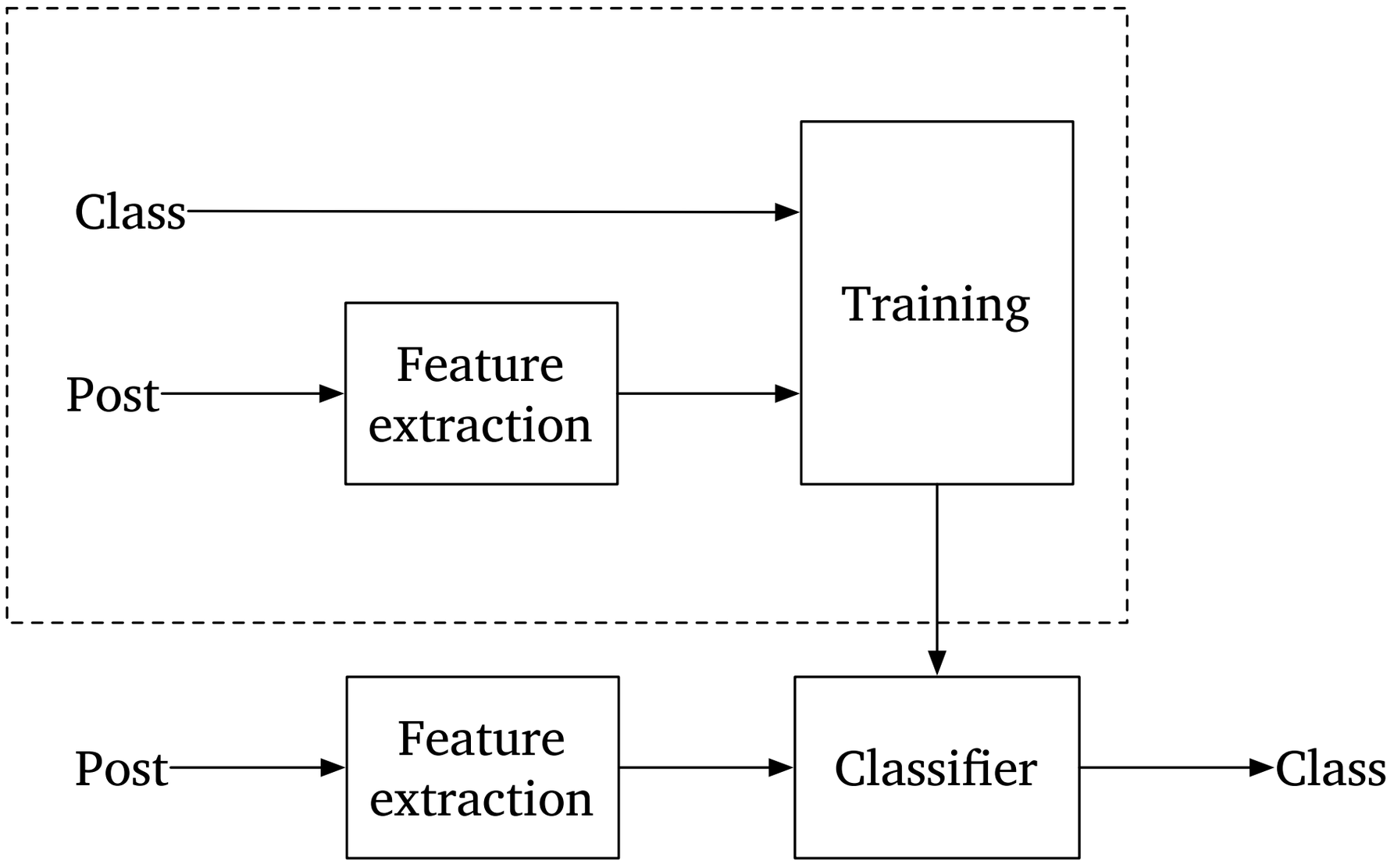
Minimum frequency cutoff

1	93,122	74.07
2	51,036	75.29
3	37,486	75.77
4	30,617	75.35
5	26,394	75.66
6	23,148	75.29
7	20,948	75.29
8	19,093	75.08
9	17,620	74.66
10	16,388	74.60
11	15,335	74.71
12	14,450	74.81
13	13,659	74.66
14	12,957	74.50
15	12,340	74.23
16	11,762	74.02
17	11,274	73.75
18	10,860	73.70
19	10,434	73.49

Feature selection

- Very large models are expensive to train, store, and apply
- Very large models also have slightly lower accuracy than pruned models
- **Overfitting –** large, complex models can capture properties of the training data that reduce training error but increase **generalization** error

Text classification



Text classification

- Construct a **document-term matrix**

```
V = CountVectorizer(…)
V.fit(training data)
X_train = V.transform(training data)
```

- Construct classifier

```
model = BernoulliNB(…)
model.fit(X_train, y_train)
```

- Apply classifier to new data

```
X_test = V.transform(test data)
y = model.predict(X_test)
```

Scikit-learn

- Consistent interface for feature extraction

  X.fit
  X.transform
  X.fit_transform

  and classifiers

  X.fit
  X.fit_predict
  X.predict

- Support for model comparison and selection (cross validation, grid search)

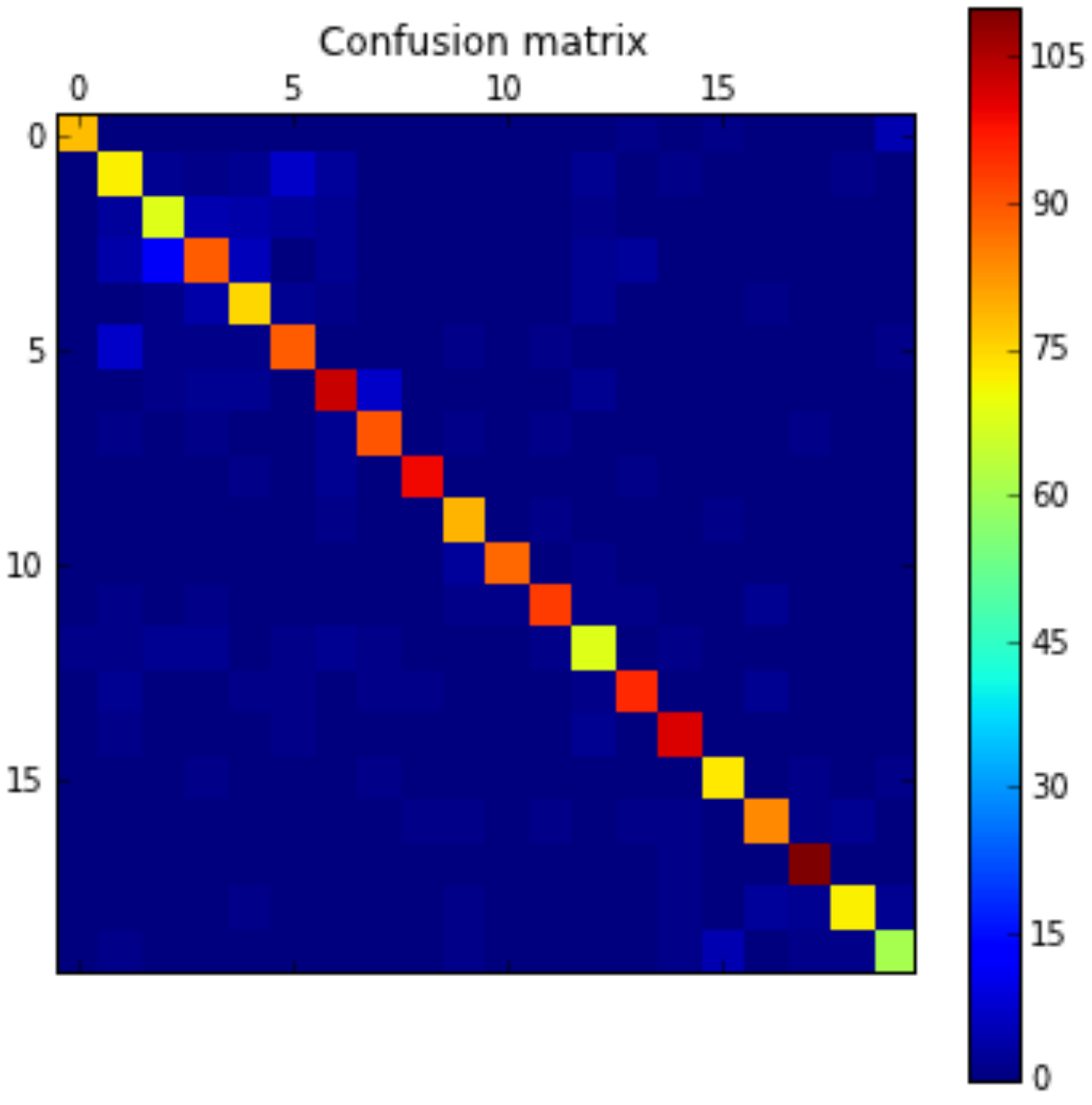
Model selection

- Using the same data, train a Linear Support Vector Machine
- Same features, training, and eval data
- Overall accuracy is 89.64%!

Error analysis

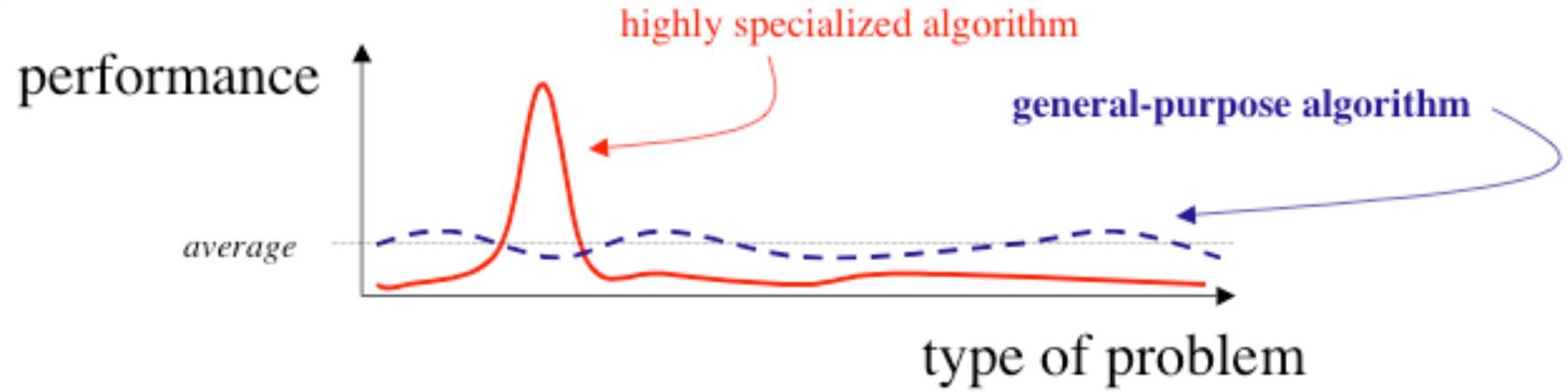
	precision	recall	f-score	support
alt.atheism	0.99	0.92	0.95	85
comp.graphics	0.77	0.79	0.78	91
comp.os.ms-windows.misc	0.78	0.79	0.79	86
comp.sys.ibm.pc.hardware	0.83	0.75	0.79	118
comp.sys.mac.hardware	0.81	0.87	0.84	86
comp.windows.x	0.86	0.87	0.86	102
misc.forsale	0.87	0.88	0.88	117
rec.autos	0.90	0.93	0.91	97
rec.motorcycles	0.98	0.96	0.97	103
rec.sport.baseball	0.90	0.96	0.93	82
rec.sport.hockey	0.99	0.96	0.97	92
sci.crypt	0.95	0.92	0.93	101
sci.electronics	0.83	0.85	0.84	80
sci.med	0.93	0.91	0.92	104
sci.space	0.94	0.96	0.95	105
soc.religion.christian	0.91	0.95	0.93	77
talk.politics.guns	0.91	0.91	0.91	92
talk.politics.mideast	0.95	0.99	0.97	111
talk.politics.misc	0.95	0.88	0.91	82
talk.religion.misc	0.87	0.86	0.87	71
avg / total	0.90	0.90	0.90	1882

Error analysis



TANSTAAFL

- **No Free Lunch Theorem** (Wolpert & Macready 1994)
  All algorithms that search for an extremum of a cost function perform exactly the same, when averaged over all possible cost functions.



- There is no "best" machine learning method *if problems are uniformly distributed*
- Then why don't we just randomly generate solutions?
- Problems aren't uniformly distributed, of course!

TANSTAAFL

- Much machine learning lore relates to the kinds of problems that particular algorithms are particularly well suited for
- Much more rarely, we see the kinds of problems that particular algorithms are particularly ill suited for
- Choosing the best method in a particular situation is often a matter of trial and error (good for thesis topics!)
- Most of the time, different methods give pretty much the same results (even better for thesis topics!)

Probabilistic models

- There are two ways of applying the Bayes decision rule
- A **discriminative** (aka **diagnostic**) method directly models $P(c|x)$
- More commonly, a **generative** (aka **sampling**) method is used, which models the joint distribution $P(x,c)$ and uses Bayes rule:

$$\begin{aligned}\hat{c} &= \underset{c \in C}{\operatorname{argmax}} P(c|x) \\ &= \underset{c \in C}{\operatorname{argmax}} \frac{P(x|c)\,P(c)}{P(x)} \\ &= \underset{c \in C}{\operatorname{argmax}} P(x|c)\,P(c) \\ &= \underset{c \in C}{\operatorname{argmax}} P(x,c)\end{aligned}$$

Naive Bayes classifiers

- To make a generative classifier, we need $P(x,c)$
- We can break this down into two parts: the **class prior** $P(c)$, and a **likelihood** $P(x|c)$
- The class priors are easy to estimate from training data:

$$\hat{P}(c) = \frac{\#\text{ of instances in class } c}{\#\text{ of instances}}$$

- This won't work for $P(x|c)$, since any particular feature vector x is unlikely to turn up in the training data

$$\hat{P}(x|c) = \frac{\#\text{ of instances of } x \text{ in } c}{\#\text{ of instances in } c} \approx \frac{0}{\#\text{ of instances of } x \text{ in } c}$$

Naive Bayes classifiers

- To get a better estimate of $P(x|c)$, we can make the simplifying assumption that each of the dimensions x_i in x are independent, so that:

$$P(x|c) = \prod_i P(x_i|c)$$

- Now we only need to get estimates of $P(x_i|c)$ from the data for each x_i, which we can do in the usual way:

$$\hat{P}(x_i|c) = \frac{\#\text{ of instances of } x_i \text{ in } c}{\#\text{ of instances in } c}$$

- Both $P(c)$ and $P(x_i|c)$ can be estimated using whatever tricks we have available

"All Fit News Print" That's the to

New The Times York

Edition Late

41, 64, 78, 86, a sail clouds, cloudy, 20. Details, fog, high high high low low mile mostly Nine of Page partly partly shower, sun to Belles, Yiddish new. Tonight, Yesterday, York.

52,847 CLIII No. VOL. ... ©2004 Copyright New The Times York 0 ... 22, 2004 MAY NEW WEDNESDAY, YORK ... DOLLAR ONE

a a a American and arms building building exploded, for in in Iraqi Kurbala, sweeps near search storming the troops were when yesterday

a Abuse Aide And Clash Gen eral on Rumsfeld

a at Hearing Is Issue M.P.'s of Role Senate

By ERIC SCHMITT

11 — a a a a a a A. Abu abuses abuses abusers against against aides also also also American Amer icans an an and and an other An tonio Armed Army Army Army Army Army's as as as as as as as as as as attacks been been better be- tween But by by by called Cambone Cam bone cases charge civilian civilian cli mated closely com- lon com- mand

com-mand, Committee conclusion conditioning conditions conditions" confusing contradicted contributed culpa- bility Defense de- fense detained details-ons directed disputed dis doctrine Donald effectively expect extract facility, full low finding first for for for for for fragmented from from from gone Gen. General General general general general general general. Gi rault-guards guards guards. H. had have He He he he hearing help how Ha his "How' if in in in inept

influenced informa- tion inquiry inquiry's insurgents intelligence in intelligence intelligence intelligence in intelligence intelligence intelligence in intelligence, interrogation, interviews gated involved Iraq Iraqi Iraqi Iraq is issue it 'can last last M. May, May military military military military military military misinterpreted M.P.'s Mr. much necessarily needed next not not not November November occur red, of of of of of of of officer officer, officers officers officers, official, on on not only order order, others, oth

AMERICAN AN DECAPITATION IRAQ OF SHOWS TAPE THE

CALLED IS KILLING REVENGE

Abuse Al by by Cited G.I.'s Is Militant Prisoner Qaeda Tied to Who

By DEXTER FILKINS

11 — 28, a a a a a a a a a A.M.] Abu Abu Abu Abu Abu Abu Al al-Anime, al-Gargawi, American American American American Americ-can American-can Americans An An an an an an an and and and attempts appeared As as as Associated at attacks

BAGHDAD, Baghdad, he hearded before he- fore," beheading behind believe Berg Berg Berg Berg, Berg, Berg's Berg's Berg's body body Bush businessman by by called com- era, carried carries Chester, commanding con- cerned

deadliest death, decapitated de capitation Deputy Det Department Department family father flew facing floor for found found Ghraib "God gone Great!" group had had had hands he he hand head, held here here, himself his "I identified in

in in in in independent infidel in intelligence Iraq Iraq, Iraq, Iraqi Iraqis is is Islamic Islamist issued Jordan ian Jump killers knew knife linked looking man men, masked May men men men men, Michael multi- nan nationals nationals nationalist Monday more."

Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mustafa Mustafa Mustafa named near neck

by Cleric's Held Iraqi Militia Mosque Strikes U.S.

and By DEXTER EDWARD FILKINS WONG

12 — 30 1-30 A a a a a a a a Abbas, Abbas, abuse age Adnan after against agencies agreed Ahiyr al-Sadr, al- Zarfi, a an, American American Am- erican American American Americ- an

emerged event

extensive fearing favored field First, First first five-week-long For exclusion forces forces forces for- ces, from fun- four fury gated gover nor graphic groups had had had had handed he he him, his his his his his' holy holy holy Hotel hundred hundreds

new new, number occupying of of of of of of of of of of of of of of of of of of of other, officers office officers On on on on on, operation or organ- izing out out out not not peace ful personnel Peter photo positions prisoners, rebel recon- ciliating rou lation run- Sadr Sadr Sadr, Sadr's said said, seized several Shiite

Abuse Account Afghan Gives Of Own U.S.

By CARLOTTA GALL

13 — 80 44 47, 2002 A a a about

Naive Bayes classifiers

- The naive Bayes classifier selects the class such that:

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} P(c) \prod_{i} P(x_i|c)$$

- Naive Bayes classifiers have been used primarily for classifying texts (Maron 1961)
- We treat a text as a **set** or **bag of words**, an unordered collection of all the words that appear in the text
- “We treat a text as a set or bag of words”
  =
  {a, a, as, bag, of, or, set, text, treat, we, words}

Naive Bayes classifiers

- Ignoring word order in the feature representation removes the most obvious syntactic dependencies between words

$$P(\text{the})\,P(\text{book}) \neq P(\text{the book})$$

- There are still semantic dependencies:

$$P(\text{tackle})\,P(\text{touchdown}) \neq P(\text{tackle}, \text{touchdown})$$

- And, multiple occurrences of words are probably not independent

Feature selection

- A straight bag-of-words model leads to positing a very large number of features
- Some of those features will not be relevant for the task (stop words)
- Many of the features will appear relevant, but won't be: we can't avoid the Curse of Dimensionality
- So, we want to select a subset of features which appear promising, usually by frequency, mutual information, or information gain

Event models

- If we represent a document as a **set** of words, then each feature x_i is a Bernoulli variable, where:

$$P(x_i|c_j) = P(x_i = 1|c_j)^{x_i} \left(1 - P(x_i = 1|c_j)\right)^{1-x_i}$$

- If there are v words in the vocabulary, a document is constructed by flipping v coins
- Call p_{ij}=$P(x_i$=1|$c_j)$. Substituting this in, we get:

$$P(c_j|x) = \frac{P(c_j)\prod_i P(x_i|c_j)}{P(x)}$$

$$= \frac{P(c_j)\prod_i p_{ij}^{x_i}(1-p_{ij})^{1-x_i}}{P(x)}$$

Event models

- Under this **binary independence model**, the parameters p_{ij} can be estimated via:

$$\hat{p}_{ij} = \frac{\#\text{ of documents containing } x_i \text{ in } c_j}{\#\text{ of documents in class } c_j}$$

- Note that this doesn't take into account the length of the document
- It also doesn't take into account the number of times a word appears in a document

Event models

- If instead we represent a document as a **bag** of words, then we can model a document as a sequence of random draws from a multinomial distribution
- The probability of picking word w_i if the document class is c_j once is $P(w_i|c_j)$
- The probability of picking word w_i x times in a row is $\mathrm{P}(w_i|c_j)^x$
- The probability of drawing a collection of words *in that order* is:

$$\prod_i P(w_i|c_i)^{x_i}$$

Event models

- This underestimates $P(x|c_j)$, since lots of ordered sequences correspond to the same bag of words
- How many different ways are there to draw word w_1 x_1 times, word w_2 x_2 times, and so on?
- We can use the **multinomial coefficient**:

$$\binom{n}{n_1, n_2, \ldots} = \binom{n}{n_1} \times \binom{n - n_1}{n_2} \times \cdots$$

$$= \frac{n!}{n_1!(n - n_1)!} \times \frac{(n - n_1)!}{n_2!(n - n_1 - n_2)!} \times \cdots$$

$$= \frac{n!}{n_1! n_2! \cdots}$$

Event models

- So, if we draw $N = \sum_i x_i$ words, we have

$$P(x|c_j) = \binom{N}{x_1, x_2, \ldots} \prod P(w_i|c_j)^{x_i}$$
$$= N! \prod \frac{P(w_i|c_j)^{x_i}}{x_i!}$$

- To be completely correct, we also need to think about the probability of finding a document of a particular length

$$P(x|c_j) = P(N|c_j)(\sum_i x_i)! \prod \frac{P(w_i|c_j)^{x_i}}{x_i!}$$

but in practice this can be hard to do

Event models

- The parameters of the multinomial model are the individual word probabilities $P(w_i|c_j)$
- Since these are the parameters of a multinomial distribution, we need to maintain:

$$\sum_i P(w_i|c_j) = 1$$

- We can estimate those from training data as:

$$\hat{P}(w_i|c_j) = \frac{\#\text{ of times } w_i \text{ occurs in documents in } c_j}{\#\text{ of words in documents in class } c_j}$$

- As always, smoothing is important

Text classification

- The multinomial model takes word frequencies and document length into account, but treats multiple occurrences of a word as independent events
- McCallum and Nigam (1998) compare the two event models
- Multinominal model almost always outperforms multivariate Bernoulli model, by 25% or so
- The multinominal model handles large vocabulary sizes much better
- It’s easier to see how to add non-text features and to account for limited inter-dependencies using a multivariate Bernoulli model

Naive Bayes classifiers

- Maron (1961):

  It is feasible to have a computing machine read a document and to decide automatically the subject category to which the item in question belongs. No real intellectual breakthroughs are required before a machine will be able to index rather well. Just as in the case of machine translation of natural language, the road is gradual but, by and large, straightforward.