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 and 2 in *Introduction to Machine Learning with Python*

As Americans mourned the death of the Rev. **Billy Graham** on Wednesday, most remembered him as a pastor with the ability to lead thousands to **Jesus**, take presidents under his wing and console a nation after the Sept. 11 terrorist attacks.

But it was in the suburbs of Chicago where he learned how to amplify his voice as a preacher.

It didn’t take long for **Graham** and the congregation of Western Springs Baptist Church, his first pulpit after graduation from Wheaton College, to conclude he was better suited to preach in stadiums than sanctuaries.

“This is where he got a taste of glory, a taste of fame and the gratification that comes from speaking to huge crowds,” said **Grant Wacker**, author of “America’s Pastor: **Billy Graham** and the Shaping of a Nation,” a biography of the national icon. “And he got the response he was looking for.”

Intel reports that it has developed a stable microcode update to address the **Spectre** flaw for its Skylake, **Kaby Lake**, and Coffee Lake processors in all their various variants.

The microcode updates help address **Spectre** variant 2 attacks. **Spectre** variant 2 attacks work by persuading a processor's branch predictor to make a specific bad prediction about which code will be executed. This bad prediction can then be used to infer the value of data stored in memory, which, in turn, gives an attacker information that they shouldn't otherwise have. The microcode update is designed to give operating systems greater control over the branch predictor, enabling them to prevent one process from influencing the predictions made in another process.

Intel's first microcode update, developed late last year, was included in system firmware updates for machines with Broadwell, **Haswell**, **Skylake**, Kaby Lake, and Coffee Lake processors. But users subsequently discovered that the update was causing systems to crash and reboot. Initially, only Broadwell and Haswell systems were confirmed to be affected, but further examination determined that Skylake, Kaby Lake, and Coffee Lake systems were rebooting, too.

On July 8, 2016 the Seattle F.B.I. announced they were ‘allocating resources dedicated to the D.B. Cooper case to other matters’. Which means they were no longer going to investigate the case. According to a Seattle Times report, the F.B.I. did qualify this statement a bit by adding that if new or compelling evidence came forward, that the Bureau would reopen the case.

But if the Seattle F.B.I. was hoping that the **Cooper** case would simply ‘go away,’ and the constant tips stop coming in, they were wrong. Seattle F.B.I. agent **Ayn Dietrich-Williams** admitted the tips just kept on coming, no matter what the F.B.I. did to try and make the public lose interest. The F.B.I. also claimed they had investigated every possible suspect over the years, and checked out all credible tips.

Strangely enough, the F.B.I. has kept other famous cases open, such as the disappearance of Jimmy Hoffa in 1975, as well as the Zodiac serial killer case that predates the **Cooper** case. Could there be a different, unsaid reason why the F.B.I. chose to close the **Cooper** case? If so, what could possibly be the reason they did?

tegretol teg tegreatol tegrotol tegretal tegratal tegratol
tegtretol tegrol tegrtol tergritol tegetrol tegotol tergetol
tegrital tegerol tagratol tegeretol tegreol tegretolcr tegertol
tegreto tegrestol tegritol tegetol tegretolxr tegeratol tregratol
tergretol tegretols tegtretols trgretol tegrotal tegtetol
tegretrol tergretal teretol tegregtol tegre tygretol tergratol
tegatol tegretaol tegratols tregretol

Machine Learning

- The field of **machine learning** studies methods for writing programs which can optimize their performance (given some metric) based on experience
- A **bigram tagger** is an example of a machine learning method, while a rule-based tokenizer isn't
- Other, more general methods for learning make fewer assumptions about the underlying concept
- Related to data mining: tell me something I didn't know (unsupervised learning)
- Lots of different machine learning methods

Classification

- Part of speech tagging is a **classification** problem: assign one or more labels L from a finite set to instances I
- Classification problems come up frequently in NLP, and can be approached probabilistically by using a model to estimate $P(I\ L)$
- Classifiers can also be built by hand, e.g., as a cascade of finite state transducers which map from I to L
- A wide range of NLP tasks can be cast as classification problems

Classification

- The classification problems which we face in CL are often very complex and poorly understood, so that neither probability models nor rules are very helpful
- Our goal: present the computer with a set of properly labeled instances, and get back a classifier
- Supervised vs. unsupervised learning
- But, is learning ever *really* unsupervised?

Classification

- Part of speech tagging, chunking, named entity recognition, word sense disambiguation
- Spelling correction
- Text classification, information retrieval, automated metadata generation, message routing
- Text segmentation, text summarization
- Adjective ordering
- Anything else?

Machine Learning

- There are slightly more machine learning techniques than there are machine learning researchers
- Most “X-based Learning” algorithms fall into a few general classes
  - **Parametric** methods use a probability distribution to find the most probable solution
  - Early **non-parametric** machine learning methods used common sense strategies, plus ad hoc heuristics (decision trees, memory based learning)
  - **Deep learning** uses gradient descent to optimize a complex objective function

Machine Learning

- Classifiers differ in the range of concepts they are capable of learning
- Parametric models, number of parameters
- Decision boundaries for non-parametric methods
- Machine learning algorithms also differ in their computational properties

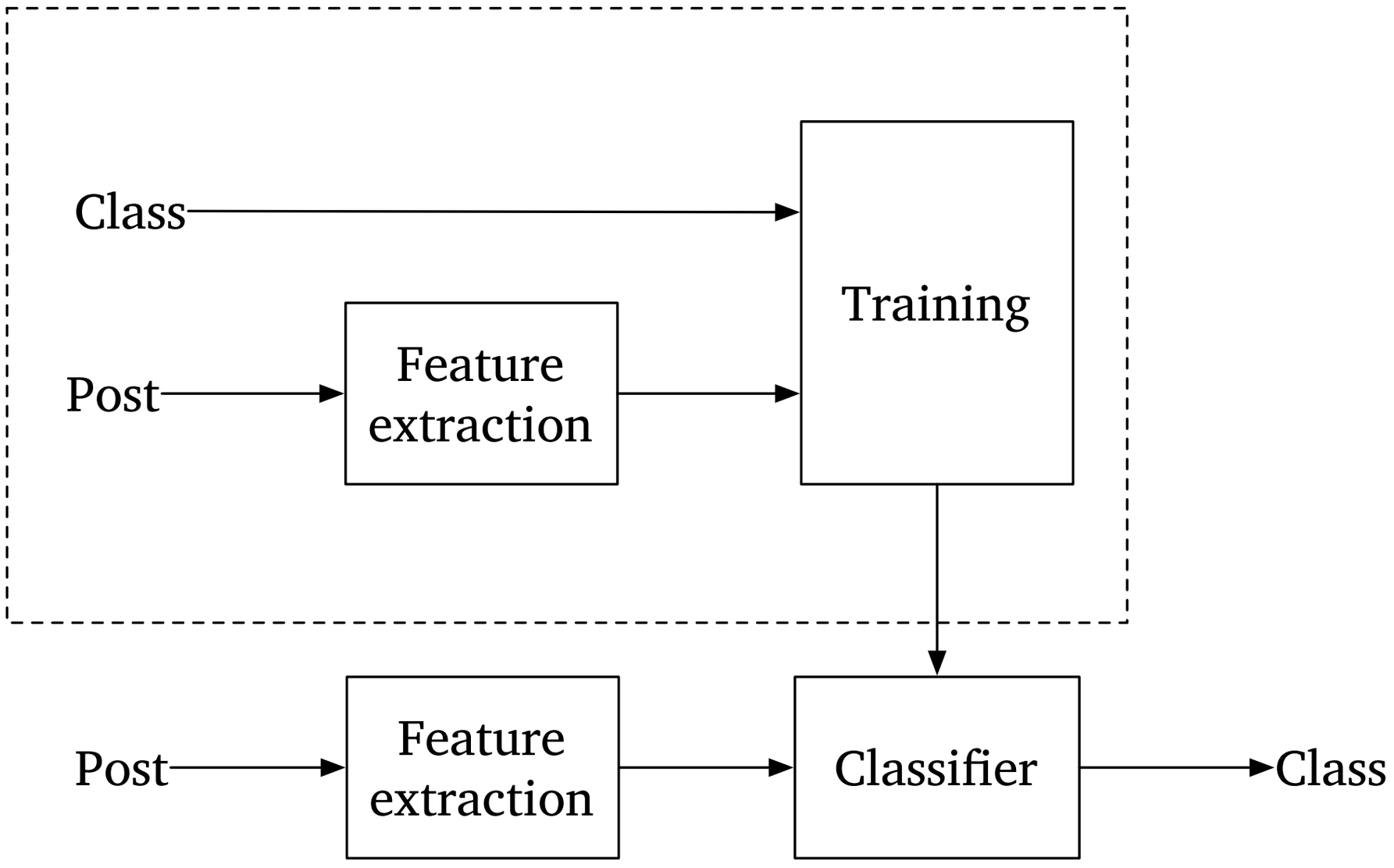
Text classification

- 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

- How well does our program guess which newsgroup a post comes from?
- We can divide our data into three parts:
  - **Training** data provides ‘experience’
  - **Evaluation** and **test** data is used to estimate how well our program performs the task
- We can judge the performance of a classifier by its **accuracy** (the fraction of instances which it correctly labels) or **error** (1–accuracy)

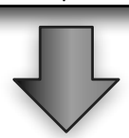
Text classification

- 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

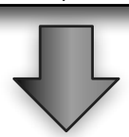
```
Say, you bought your Saturn at $13k, with a dealer profit of $2k.
If the dealer profit is $1000, then you would only be paying $12k for
the same car.  So isn't that saving money?

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 :
```



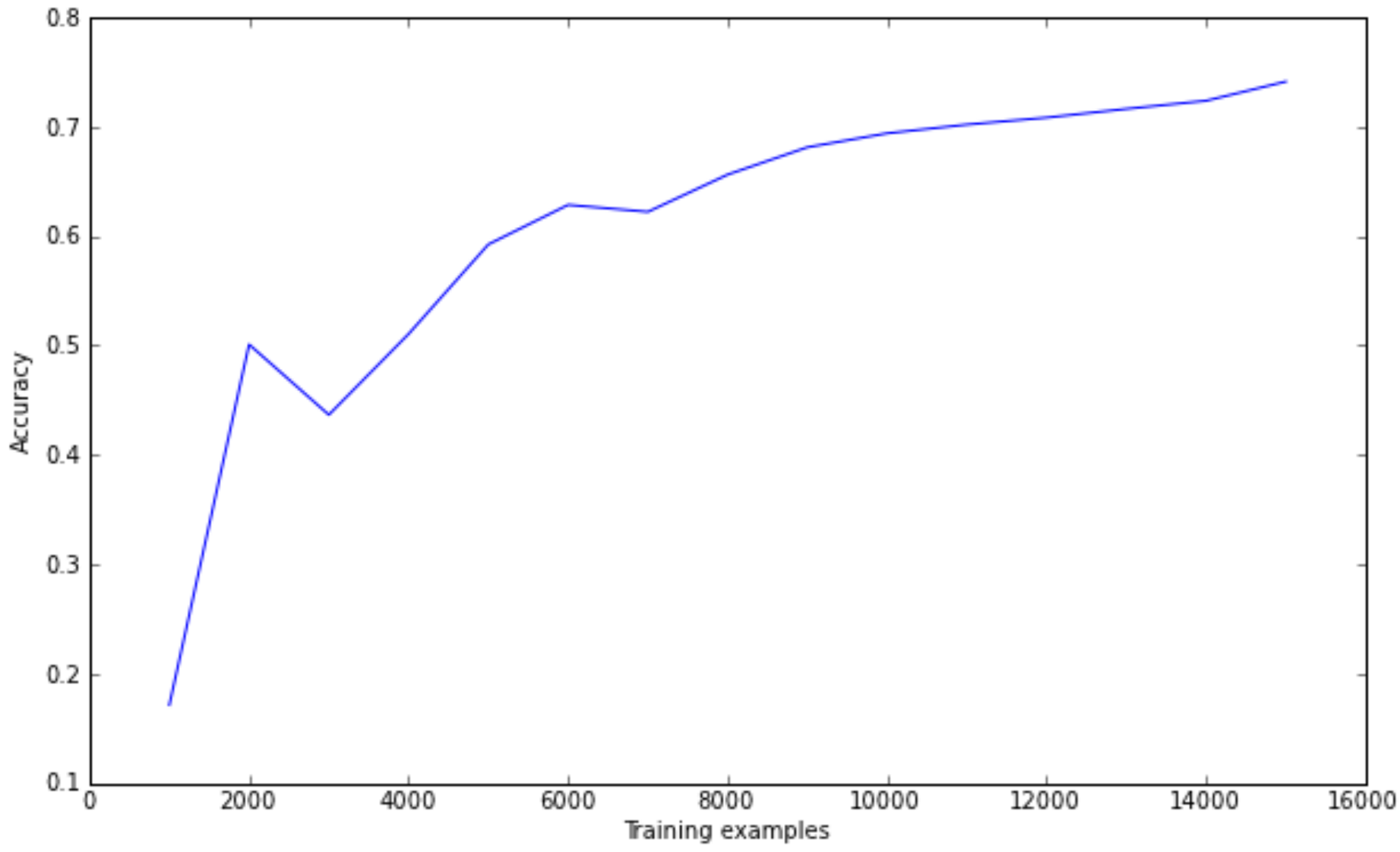
```
a after already at average be below
better bought by car cars class dealer
deals does even for if is isn it
margin market money moreover of only
paying price profit really reduce same
saturn saving say so that the their
then they was will with would you your
```

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

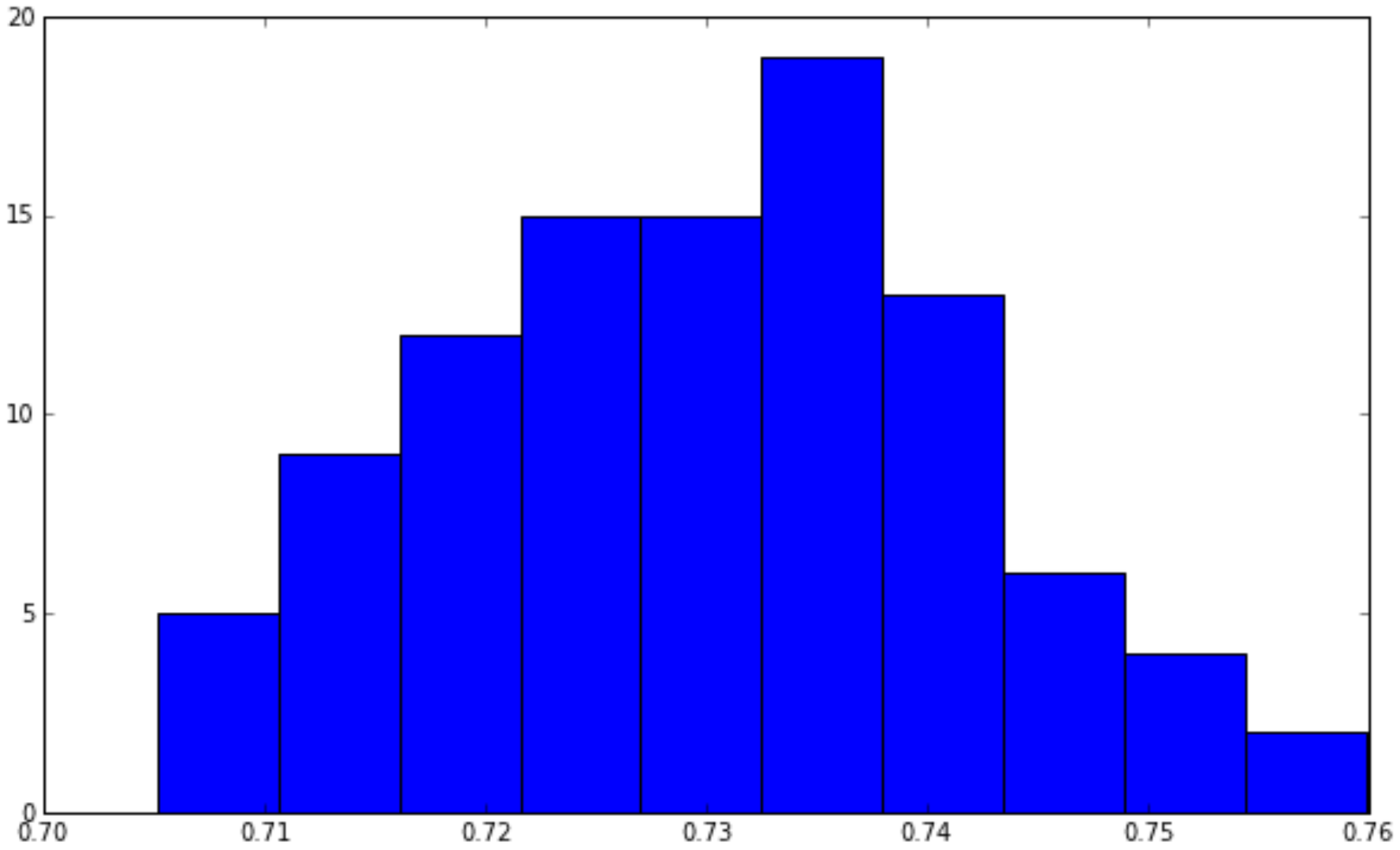
- Learning curve: performance (mostly) increases with more



- Training on all the data yields 81.35%

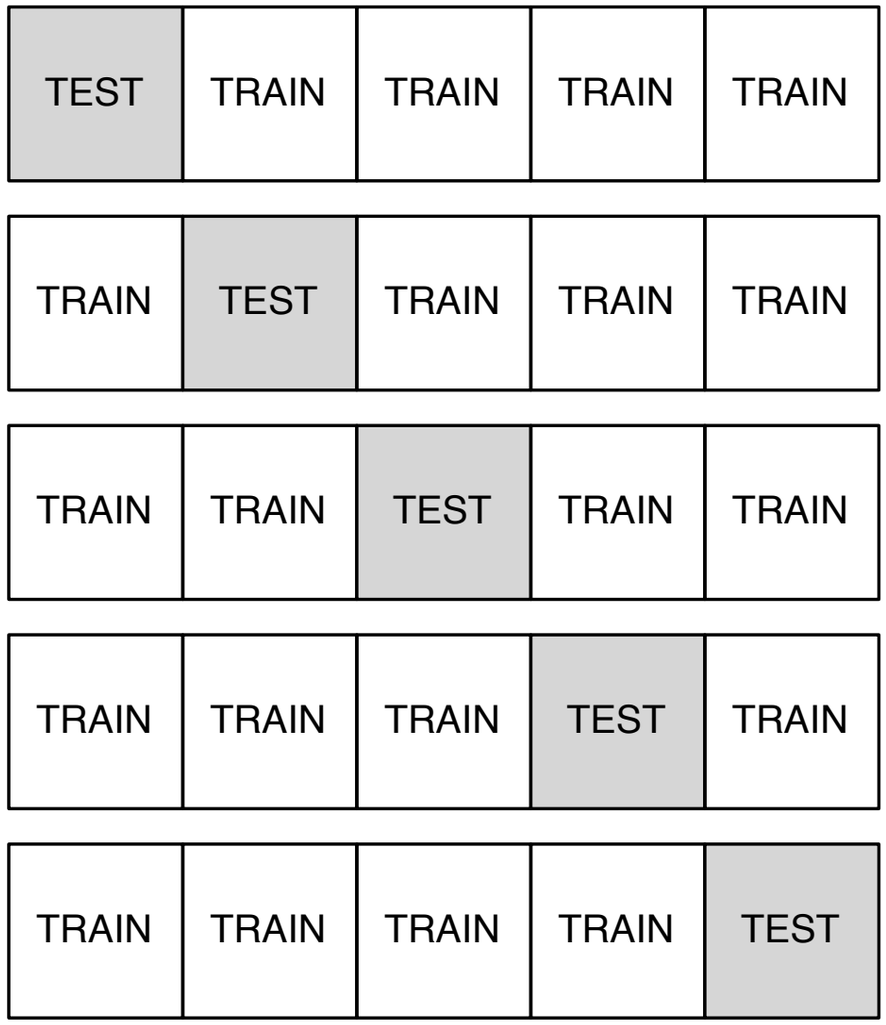
Evaluation

- 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

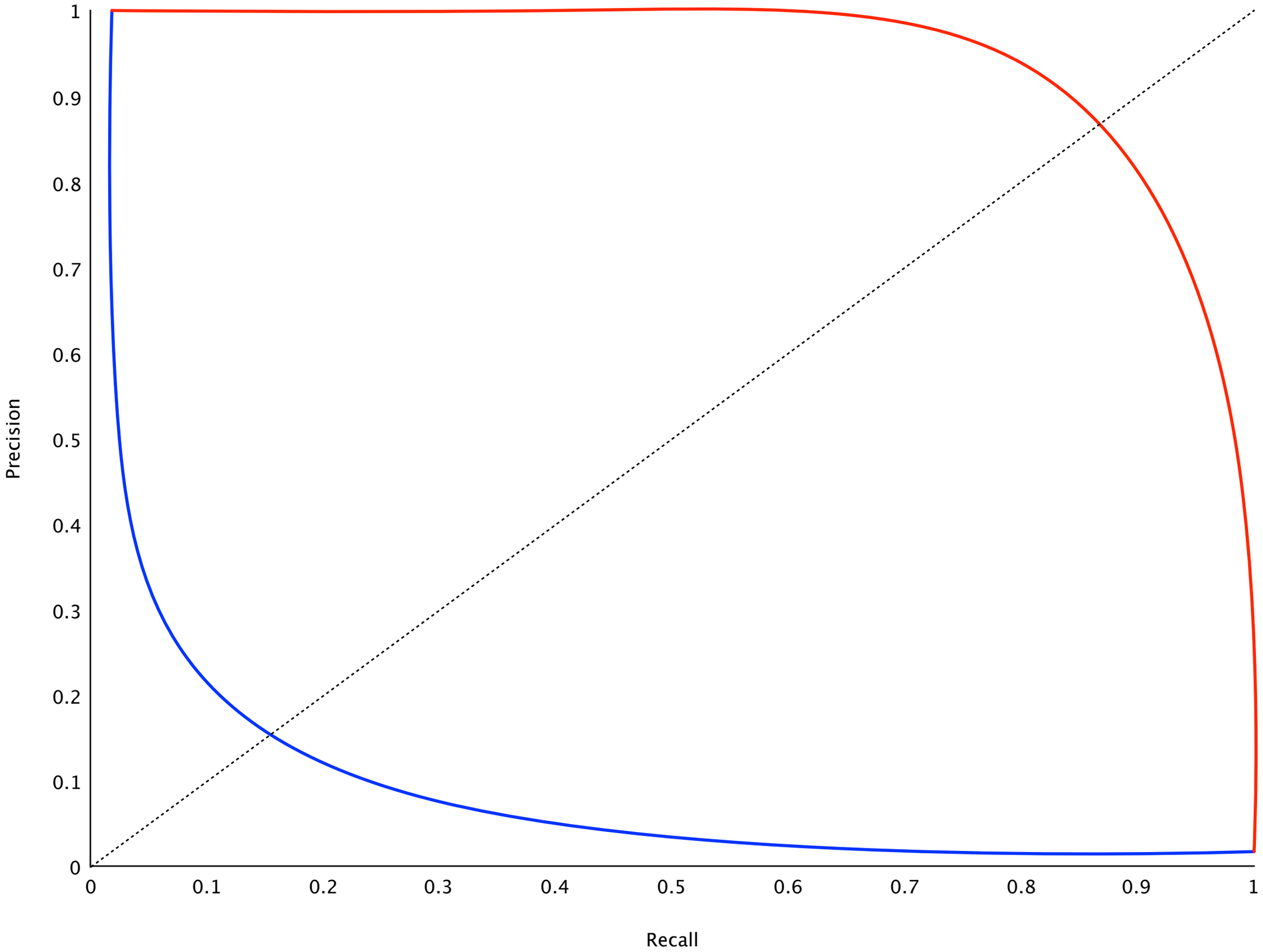
- We can judge the performance of a classifier by its **accuracy** (the fraction of instances which it correctly labels) or error (1–accuracy)
- For each class, we count false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN)

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

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

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

Evaluation


Precision
Recall
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Evaluation

- Since precision and recall involve tradeoffs, we sometimes combine them into a composite score (F score, breakeven)
- **Utility** is the most general metric, with arbitrary weights for different kinds of errors
- For multi-class problems, an overall score can be computed either by microaveraging (by instance) or macroaveraging (by class)

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