Natural Language Processing

CSCI 5832—Lecture 7

Jim Martin

Today

- Text classification (Chapter 4)
  - Practical Classification tasks
    - Spam detection, author identification, intent, topic detection
  - Naïve Bayes
    - Class-based unigram language model
  - Evaluation of Classifiers
  - Significance Testing

Is This Spam?

Dear Friend,

I am glad to know you, but God knows you better and he knows why he has directed me to you at this point in time so do not be surprise at all. My names are Mrs. Donna Louise McInnes a widow, i have been suffering from ovarian cancer disease. At this moment i am about to end the race like this because the illness has gotten to a very bad stage, without any family members and no child. I hoped that you will not expose or betray this trust and confident that I am about to entrust on you for the mutual benefit of the orphans and the less privileges ones. I have some funds I inherited from my late husband, the sum of ($11.000.000 Eleven million dollars.) deposited in the Bank. Having known my present health status, I decided to entrust this fund to you believing that you will utilize it the way i am going to instruct herein.

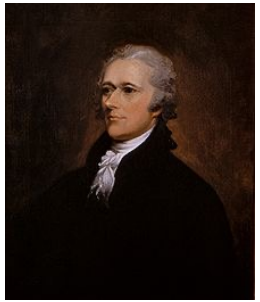
It will be my great pleasure to compensate you with 35 % percent of the total money for your personal use, 5 % percent for any expenses that may occur during the international transfer process while 60% of the money will go to the charity project.

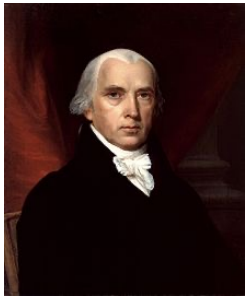
All I require from you is sincerity and ability to complete God task without any failure. It will be my pleasure to see that the bank has finally release and transfer the fund into your bank account therein your country even before I die here in the hospital, because of my present health status everything need to be process rapidly as soon as possible. I am waiting for your immediate reply, if only you are interested for further details of the transaction and execution of this charitable project.

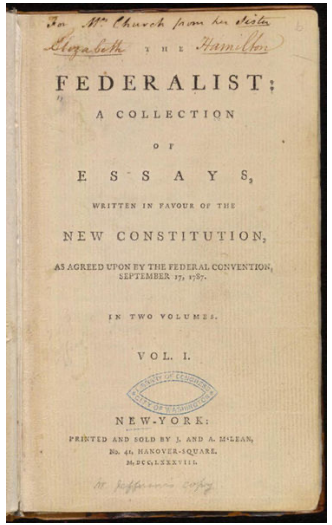
Best Regards your friend Mrs.
Donna Louise McInnes.

Who Wrote Which Federalist Paper?

- 1787-8: Anonymous essays written to convince New York to ratify new U.S. Constitution: Jay, Madison, Hamilton.
- Authorship of 12 of the letters in dispute
- 1963: solved by Mosteller and Wallace using Bayesian methods

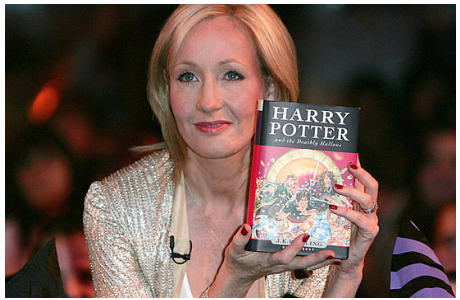


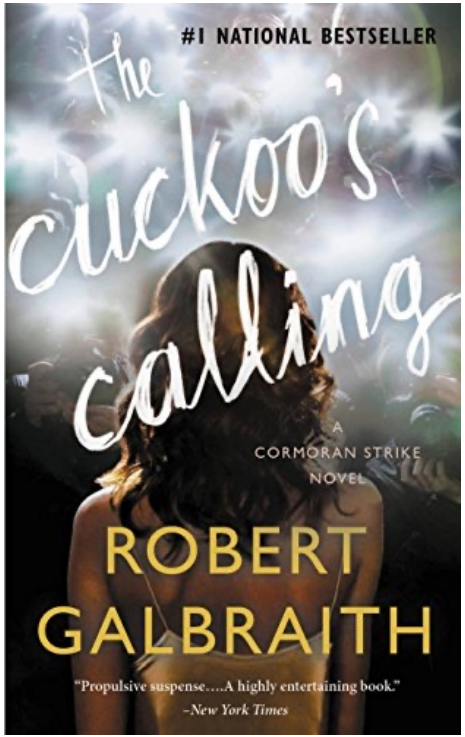




More Recent Examples

- *The Cuckoo's Calling* by Robert Galbraith





More Recent Examples

- *NY Times "Anonymous" editorial*

The New York Times

I Am Part of the Resistance Inside the Trump Administration

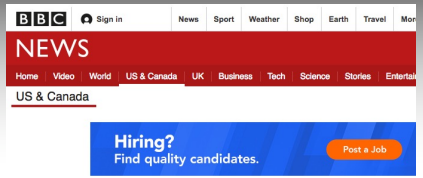
I work for the president but like-minded colleagues and I have vowed to thwart parts of his agenda and his worst inclinations.

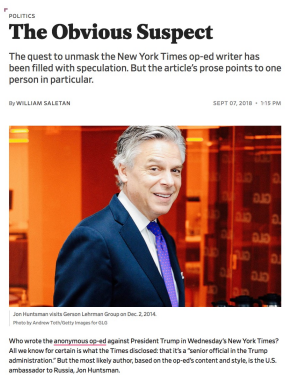
Sept. 5, 2018

The Times is taking the rare step of publishing an anonymous Op-Ed essay. We have done so at the request of the author, a senior official in the Trump administration whose identity is known to us and whose job would be jeopardized by its disclosure. We believe publishing this essay anonymously is the only way to deliver an important perspective to our readers. We invite you to submit a question about the essay or our vetting process here. [Update: Our answers to some of those questions are published here.]

President Trump is facing a test to his presidency unlike any faced by a modern American leader.

It's not just that the special counsel looms large. Or that the country is bitterly divided over Mr. Trump's leadership. Or even that his party might well lose the House to an opposition hellbent on his downfall.





Movie Review: Positive or Negative?

- *unbelievably disappointing*

- *Full of zany characters and richly applied satire, and some great plot twists*

- *this is the greatest screwball comedy ever filmed*
- *It was pathetic. The worst part about it was the boxing scenes.*

Multinomial: Article Topics

Journal Article

Syntactic frame and verb bias in aphasia: Plausibility judgments of undergoer-subject sentences

?

MeSH Subject Category Hierarchy

- Antogonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- …

Text Classification

- *Input*:
  - document d
  - a fixed set of classes $C = \{c_1, c_2, \ldots, c_J\}$

- *Output*: A predicted class $c \in C$

Supervised Machine Learning

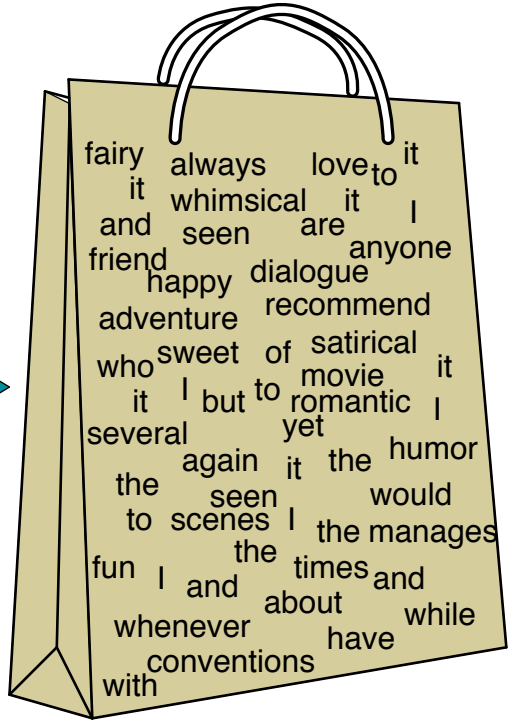
- *Input:*
  - A document *d*
  - A fixed set of classes $C = \{c_1, c_2, \ldots, c_J\}$
  - A training set of *m* hand-labeled documents $(d_1, c_1), \ldots, (d_m, c_m)$
- *Output:*
  - A learned classifier $h(d) \rightarrow c$

Naïve Bayes Intuition

- Classification method based on Bayes rule and some naïve independence assumptions
  - This should sound familiar
- When applied to text it relies on very simple representation of documents
  - Bag of words

Bag of Words

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
…	…

Naïve Bayes Classifier (1)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(c \mid d)$$

MAP is "maximum a posteriori" = most likely class

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

Bayes Rule

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

Dropping the denominator

Naïve Bayes Classifier (2)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(d \mid c) P(c)$$

$$= \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

Document d represented as features x1..xn

Naïve Bayes Classifier (3)

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

O(|X|^n•|C|) parameters

How often does this class occur?

We can just count the relative frequencies in a training corpus

This could be estimated directly only if a very, very large number of training examples was available.

Naïve Bayes Independence Assumptions

$$P(x_1, x_2, \ldots, x_n \mid c)$$

- Bag of Words assumption: Sequence doesn't matter
- Features are just word occurrences
- Conditional Independence: Assume the feature probabilities $P(x_i \mid c_j)$ are independent given the class c.

$$P(x_1, \ldots, x_n \mid c) = P(x_1 \mid c) \bullet P(x_2 \mid c) \bullet P(x_3 \mid c) \bullet \ldots \bullet P(x_n \mid c)$$

Multinomial Naïve Bayes Classifier

$$c_{MAP} = \underset{c \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c) P(c)$$

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}} P(c_j) \prod_{x \in X} P(x \mid c)$$

Applying Naive Bayes to Text Classification

positions ← all word positions in test document

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j) \prod_{i \in positions} P(x_i \mid c_j)$$

Learning a Naïve Bayes Model

- First attempt: Maximum likelihood estimates
  - Simply use the frequencies in the data

$$\hat{P}(c) = \frac{N_c}{N_{doc}}$$

$$\hat{P}(w_i|c) = \frac{count(w_i, c)}{\sum_{w \in V} count(w, c)}$$

Parameter Estimation

$$\hat{P}(w_i|c) = \frac{count(w_i, c)}{\sum_{w \in V} count(w, c)}$$

Fraction of times word w_i appears among all words in documents of topic c_j

- Create one large document for topic *j* by concatenating all docs in this topic
  - Use frequency of *w* in this mega-document

Problems

- What if we have no training documents with the word *fantastic* and classified in the topic positive?

$$\hat{P}(\text{“fantastic”}|\text{positive}) = \frac{count(\text{“fantastic”}, \text{positive})}{\sum_{w \in V} count(w, \text{positive})} = 0$$

- Zero probabilities cannot be conditioned away, no matter the other evidence!

Laplace Smoothing for Naïve Bayes

$$\hat{P}(w_i|c) = \frac{count(w_i,c)+1}{\sum_{w\in V}(count(w,c)+1)} = \frac{count(w_i,c)+1}{\left(\sum_{w\in V} count(w,c)\right)+|V|}$$

Practical Issues

- Avoiding underflow
  - Multiplying lots of probabilities can result in floating-point underflow.
  - $\log(xy) = \log(x) + \log(y)$
    - Sum logs of probabilities instead of multiplying probabilities.
    - Class with highest un-normalized log probability score is still most probable.

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

Machine Learning

- This model is now essentially a weighted sum of features and a class bias term

$$c_{NB} = \underset{c_j \in C}{\operatorname{argmax}} \log P(c_j) + \sum_{i \in positions} \log P(x_i \mid c_j)$$

- You can think of this as a scoring function
- Moving on, we can move away from just word-based features and find better ways to set the weights

Training

function TRAIN NAIVE BAYES(D, C) **returns** log $P(c)$ and log $P(w|c)$

for each class $c \in C$ # Calculate $P(c)$ terms

N_{doc} = number of documents in D

N_c = number of documents from D in class c

$logprior[c] \leftarrow \log \frac{N_c}{N_{doc}}$

$V \leftarrow$ vocabulary of D

$bigdoc[c] \leftarrow$ **append**(d) **for** d $\in$ D **with** class c

for each word w in V # Calculate $P(w|c)$ terms

$count(w,c) \leftarrow$ # of occurrences of w in $bigdoc[c]$

$$loglikelihood[w,c] \leftarrow \log \frac{count(w,c) + 1}{\sum_{w' \text{ in } V} (count(w',c) + 1)}$$

return *logprior*, *loglikelihood*, V

Inference

function TEST NAIVE BAYES(*testdoc*, *logprior*, *loglikelihood*, C, V) **returns** best *c*

for each class $c \in C$
  $sum[c] \leftarrow logprior[c]$
  for each position *i* in *testdoc*
    $word \leftarrow testdoc[i]$
    if $word \in V$
      $sum[c] \leftarrow sum[c] + loglikelihood[word,c]$
return $\text{argmax}_c\ sum[c]$

Example

- Training set

	Cat	Documents
Training	-	just plain boring
	-	entirely predictable and lacks energy
	-	no surprises and very few laughs
	+	very powerful
	+	the most fun film of the summer

Example

$$P(-) = P(-)P(S|-) = \frac{3}{5} \times \frac{2 \times 2 \times 1}{34^3} = 6.1 \times 10^{-5}$$

$$P(+)P(S|+) = \frac{2}{5} \times \frac{1 \times 1 \times 2}{29^3} = 3.2 \times 10^{-5}$$

$$P(\text{“predictable”}|-) = \frac{1+1}{14+20} \qquad P(\text{“predictable”}|+) = \frac{0+1}{9+20}$$

$$P(\text{“no”}|-) = \frac{1+1}{14+20} \qquad P(\text{“no”}|+) = \frac{0+1}{9+20}$$

$$P(\text{“fun”}|-) = \frac{0+1}{14+20} \qquad P(\text{“fun”}|+) = \frac{1+1}{9+20}$$

Naïve Bayes and Language Modeling

- In general, naïve Bayes classifiers use all sorts of additional features, not just the words
  - Author ID tends to focus on use patterns of common words
  - Email spam detectors use
    - URL, email address, dictionaries, network features, time of day
- If we use only word-based features then
  - Naïve Bayes has an specific connection to language modeling.

Class Conditional Unigram Language Model

Class *+*

0.1	I
0.1	love
0.01	this
0.05	fun
0.1	film

I	love	this	fun	film
0.1	0.1	.05	0.01	0.1

P(s | +) = 0.0000005

Naïve Bayes as a Language Model

- Which class assigns the higher probability to s?

Model pos	
0.1	I
0.1	love
0.01	this
0.05	fun
0.1	film

Model neg	
0.2	I
0.001	love
0.01	this
0.005	fun
0.1	film

I	love	this	fun	film
0.1	0.1	0.01	0.05	0.1
0.2	0.001	0.01	0.005	0.1

$$P(s|pos) > P(s|neg)$$

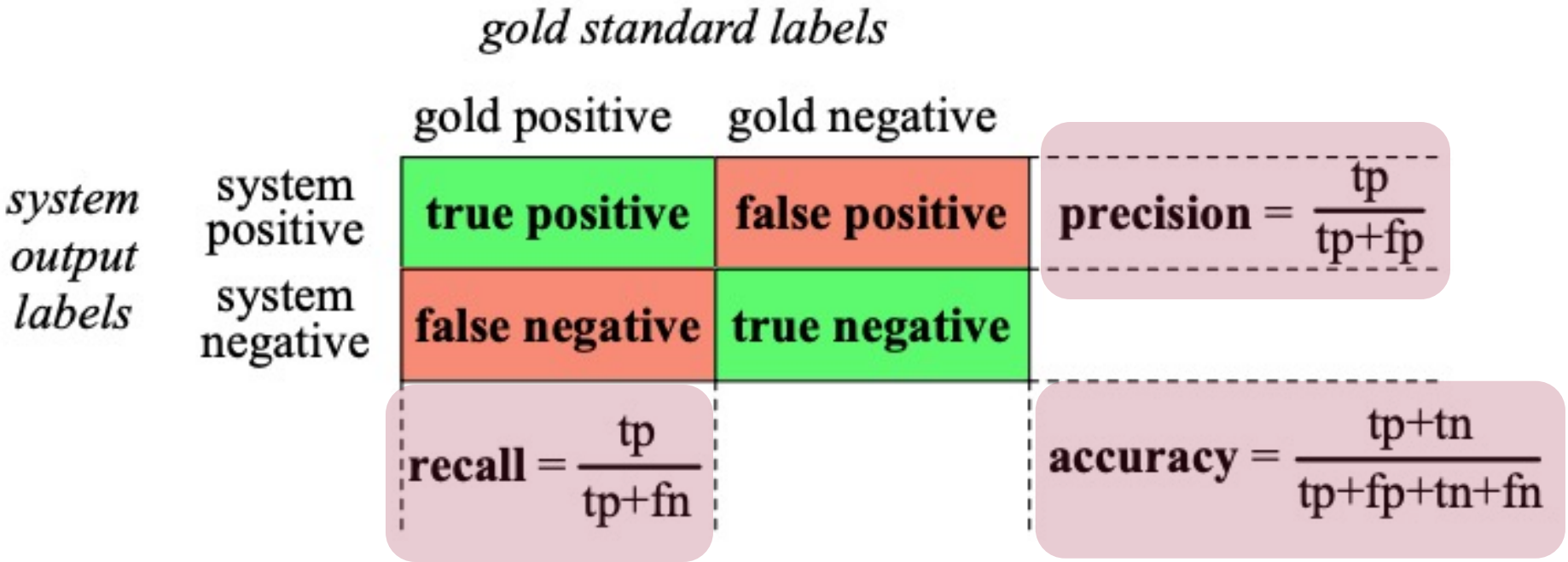
Evaluation

- How do we know how good our systems are?
- Think about the possible outcomes for an email spam detector

Evaluation

- How do we know how good our systems are?
- Think about the possible outcomes for an email spam detectors
  - Actual spam email is blocked
  - Legit email is allowed
  - Spam email gets through
  - Legit email is blocked

2-by-2 Contingency Table



Non-Binary Problems

- What about when we have more than two classes?

		gold labels urgent	normal	spam	
system output	urgent	8	10	1	$\text{precision}_u = \frac{8}{8+10+1}$
	normal	5	60	50	$\text{precision}_n = \frac{60}{5+60+50}$
	spam	3	30	200	$\text{precision}_s = \frac{200}{3+30+200}$
		$\text{recall}_u = \frac{8}{8+5+3}$	$\text{recall}_n = \frac{60}{10+60+30}$	$\text{recall}_s = \frac{200}{1+50+200}$	

A Combined Measure: F

- A combined measure that assesses the P/R tradeoff is F measure (weighted harmonic mean):

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{\beta^2 P + R}$$

- People usually use balanced F1 measure
  - i.e., with $\beta = 1$ (that is, $\alpha = ½$):

$$F_1 = \frac{2PR}{P+R}$$

Micro- vs. Macro-Averaging

- If we have more than one class, how do we combine multiple performance measures into one quantity?
  - Macro-averaging: Compute performance for each class, then average.
  - Micro-averaging: Collect decisions for all classes, compute contingency table, evaluate.

Micro- vs. Macro-Averaging: Example

Class 1

	Truth: yes	Truth: no
Classifier: yes	10	10
Classifier: no	10	970

Class 2

	Truth: yes	Truth: no
Classifier: yes	90	10
Classifier: no	10	890

Micro Ave. Table

	Truth: yes	Truth: no
Classifier: yes	100	20
Classifier: no	20	1860

- Macroaveraged precision: (0.5 + 0.9)/2 = 0.7
- Microaveraged precision: 100/120 = .83
- Microaveraged score is dominated by score on common classes

Development Test Sets and Cross-validation

- Metric: P/R/F1 or Accuracy
- Unseen test set
  - avoid overfitting ('tuning to the test set')
  - more realistic estimate of performance
- Cross-validation over multiple splits
  - Handle sampling errors from different datasets
  - Pool results over each split
  - Compute pooled dev set performance