



BC COMS 2710:
Computational Text Analysis

**Lecture 15 – Machine Learning:
Text Classification (Naive Bayes)**



- Readings 04:
 - link posted to course site
 - due Sunday

- HW 02:
 - Due Wednesday night (last night)

- HW 03:
 - Released today
 - Due next Wednesday night



- **Project ideation – Friday May 28st**
 - <https://www.overleaf.com/read/yzpgxcgsqdvj>
- roughly 250 word overview of what you are interested in

Final Project – Deliverables



- Project ideation – Friday May 28st
 - 5 points
- Project proposal – ~~Friday June 4th~~ Sunday June 6th
 - 9 points
- Project presentations – Monday June 14th
 - 6 points
- Project submissions – Friday June 18th
 - 15 points
- http://coms2710.barnard.edu/final_project



When computing the same thing across a row or column, what should we do?

1. Define a function
2. apply the function

Looping through a dataframe is not ideal



“A computer program does what you tell it to do, not what you want it to do.”

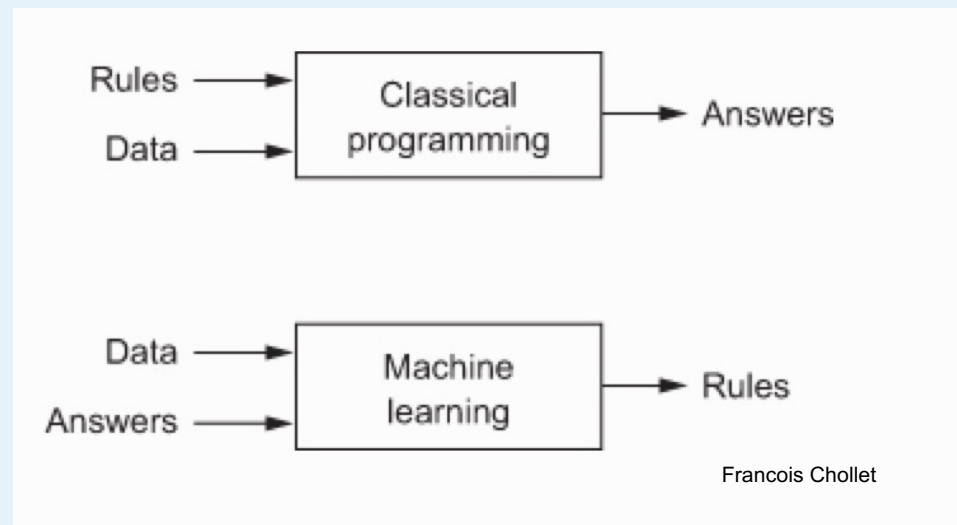
Be careful when looping and adding to lists



Machine Learning

A mathematical model
calculated based on sample data ("training data")

makes predictions or decisions without being
explicitly programmed to perform the task





- Supervised Learning
 - Learn rule from data and answers
- Unsupervised Learning
 - Learn a rule for patterns from data
- Reinforcement Learning
 - try your rule on a piece of data, and get feedback on how good your rule was

A blue-tinted photograph of a statue, likely a personification of Truth or Justice, holding a torch aloft in its right hand. The statue is set against a background of trees and a clear sky. The word "Prediction" is overlaid in a large, white, sans-serif font in the center of the image. Two short white horizontal lines are positioned above and below the text.

Prediction



- Based on incomplete information
- One way of making predictions:
 - To predict an outcome for an individual,
 - find others who are like that individual
 - and whose outcomes you know.
 - Use those outcomes as the basis of your prediction.



Two types of predictions: Classification & Regression

Classification = Categorical

Regression = Numeric

Predicting sentiment:

- Classification



- Regression:

$[-1, \dots, 1]$

Prediction Example: Hot dog or not Hot dog?





Text Classification

Spam or Not Spam?



David, Adam 6

Tennis this week? - in playing tennis on Tuesday. It >>>> will b...

Citi Alerts

Your Citibank account statement is available online - com to y...

Humane Rescue Allia.

Your HRA E-Newsletter - Read news and events updates from ...

SLEEP NUMBER

Check out these limited-time Weekend Specials - PLUS get fre...

aishagaddafi11119

Inquiry for Investment. - Inquiry for Investment. Assalamu Alai...

What is this medical article about?



MEDLINE Article

Available on the at www.sciencedirect.com

Brain
Cognition
www.elsevier.com/locate/brain

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

Susanne Gahl,^a Lisa Mann,^b Gill Ramchand,^b Daniel S. Jurafsky,^b Elizabeth Elder,^b Molly Kavage,^a and L. Holland Audry^a

^a University of California, Berkeley, CA, USA
^b University of Toronto, Toronto, ON, Canada
September 1, 2010

Abstract

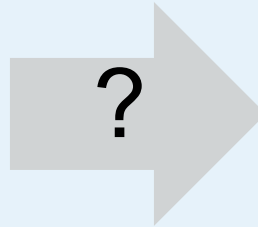
This study investigates how factors that have been argued to define “essential form” in sentence comprehension (syntactic structure, semantic role, and frequency of usage) affect the accuracy of plausibility judgments of underdog-subject sentences. Using a plausibility judgment task, we show that a small group of aphasic participants reliably forms an association between passive, for the most part, and the observation that passives are generally harder than actives for aphasia. We show that this effect is mediated by lexical bias, i.e., the likelihood that a verb appears in a given syntactic structure. Pairs of passive/active verbs were significantly easier than pairs of underdog verbs. More generally, we show that structure makes the lexical bias of the verb in a sentence reliably predict the accuracy of which structure and level has to be chosen. These findings suggest that “essential form” always happens and causal links.

© 2010 Elsevier Inc. All rights reserved.

1. Introduction

The simplicity of “essential form,” or “essential word order,” for normal and aphasic comprehension has often been taken as evidence for the sentence comprehension impairment (SCHI) as has been pointed out by Mann (2002), the privileged status of essential form (EF) verb separations. Different definitions of “essential form” yield widely different predictions. One approach to the definition of essential sentence form is that implicit in Bates, Prizack, and Saffran (1992, hereafter Bates et al.), based on a case that associates with *Agent-Patient-Object* order: represent the essential word order for English. A second approach is based on syntactic “movement” analysis and defines as crucial, hence, any word order that diverges from the [NP]₁[Verb]₂[NP]₃ configuration assumed for the core structure of English sentences. Based on this analysis, the meaning of essentiality, King (1988) argues that sentences with unaccusative verbs should be difficult to process for aphasic patients, in particular for patients with “agglutination,” for reasons that are analogous to the factors giving rise to the greater difficulty of passive compared to active. Although the precise definition of essentiality is contested (see e.g., Levin & Rappaport Hovav, 1995), unaccusative verbs are generally unable to be intransitive verbs whose (surface) subjects represent Underdog arguments. Examples of unaccusative verbs include verbs like *fall* and *blow*. Under the conventional analysis assumed in King (1988), the surface subjects of unaccusative verbs are linked via movement to their original in deep structure. Unaccusative verbs therefore inherit the very same difficulties as passive sentences, according to King’s analysis, and should be as hard as passives for aphasic speakers.

A different approach to essential form has been proposed by Mann et al. (1998) who suggest that essential form refers to the most frequent syntactic form for a given verb. Under this view, aphasic problems with processing and understanding passives derive from the fact that, for most transitive verbs, passives occur less frequently than actives. One prediction of this approach, also advanced by Gahl (2010), is that comprehension difficulty should vary with the lexical bias of the verbs



MeSH Subject Category Hierarchy

- Antagonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- ...



- + *...zany characters and richly applied satire, and some great plot twists*
- *It was pathetic. The worst part about it was the boxing scenes...*
- + *...awesome caramel sauce and sweet toasty almonds. I love this place!*
- *...awful pizza and ridiculously overpriced...*



- *Movie*: is this review positive or negative?
- *Products*: what do people think about the new iPhone?
- *Public sentiment*: how is consumer confidence?
- *Politics*: what do people think about this candidate or issue?
- *Prediction*: predict election outcomes or market trends from sentiment



Input:

- a document d
- a fix set of classes $C = \{c_1, c_2, \dots, c_n\}$
- A training set of n labeled documents
 $(d_1, c_1), (d_2, c_2), \dots, (d_n, c_n)$

Output:

- A learned classifier f
 - f is a mapping from $d \rightarrow c$



—

Classifiers

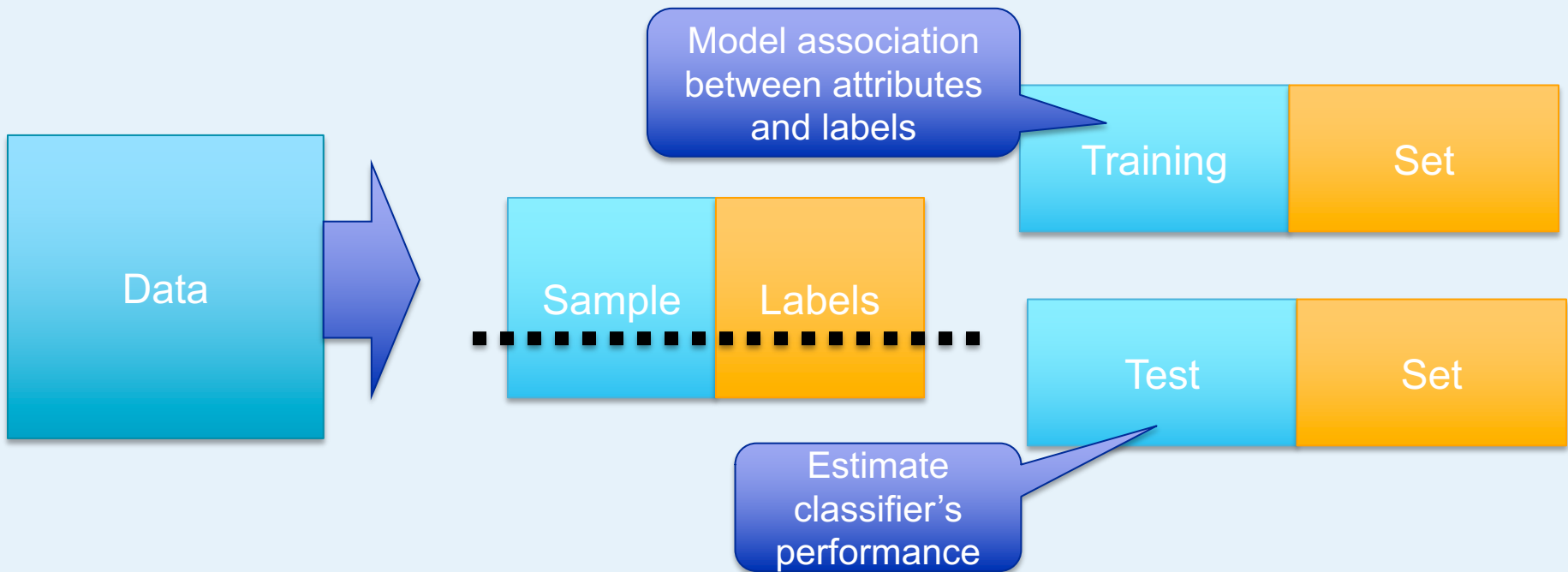
—

Attributes
(features) of
an example



Predicted
label of the
example

Setup for training and evaluating a classifier





—

Scikit-Learn

—



scikit-learn uses a standard set of functions for all models

The two main ones for our purposes

`model.fit(X, y)` — train the model with the given data set

`model.predict(X_test)` — get predictions for the given test set



- Neural Networks
- K-Nearest Neighbors
- Logistic Regression
- Naive Bayes
-

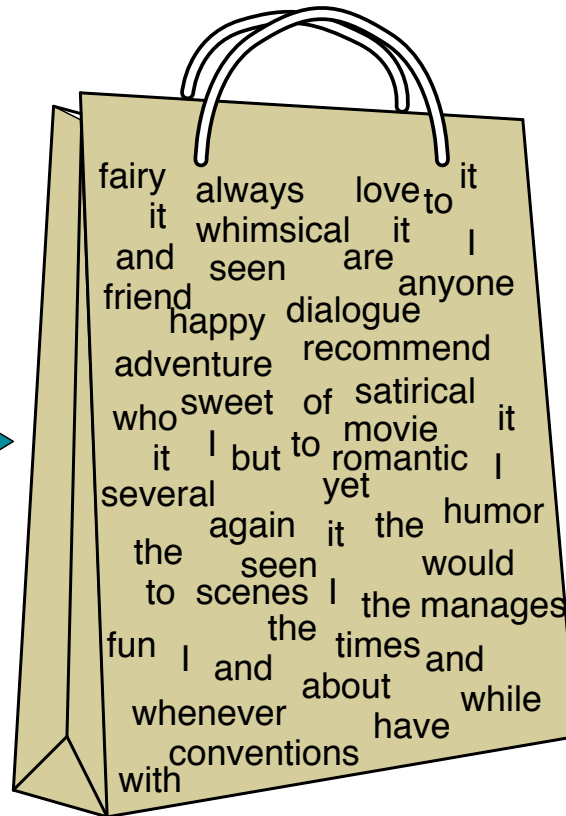


— Naive Bayes —

Bag of Words Representation



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
...	...

Classify document based on BoW



What is the probability of the class given the BoW

$$f\left(\begin{array}{|c|c|} \hline \text{seen} & 2 \\ \hline \text{sweet} & 1 \\ \hline \text{whimsical} & 1 \\ \hline \text{recommend} & 1 \\ \hline \text{happy} & 1 \\ \hline \dots & \dots \\ \hline \end{array}\right) = C$$




$$P(c | d) = \frac{P(d | c)P(c)}{P(d)}$$

Bayes Rule Derivation





Given document d , what is the probability of category c

$$P(c | d) = \frac{P(d | c)P(c)}{P(d)}$$

Naive Bayes Classifier



Choose category c that has the highest probability given document d

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

MAP is “maximum a posteriori” = most likely class

$$= \operatorname{argmax}_{c \in C} \frac{P(d | c)P(c)}{P(d)}$$

Bayes Rule

$$= \operatorname{argmax}_{c \in C} P(d | c)P(c)$$

Dropping the denominator

Choose category c that has the highest probability given document d

"Likelihood"

"Prior"

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

How do we represent document d

Answer: Bag of Words

$$= \operatorname{argmax}_{c \in C} P(x_1, x_2, \dots, x_n | c)P(c)$$



$$P(x_1, x_2, \dots, x_n | c)$$

- **Bag of Words assumption:** Assume position doesn't matter
- **Conditional Independence:** Assume the probabilities $P(x_i | c_j)$ are independent given the class c .

$$P(x_1, \dots, x_n | c) = P(x_1 | c) \cdot P(x_2 | c) \cdot P(x_3 | c) \cdot \dots \cdot P(x_n | c)$$

Plugging this into our prediction equation:

$$c_{MAP} = \operatorname{argmax}_{c \in C} P(x_1, x_2, \dots, x_n | c) P(c)$$

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



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

Count Frequencies in training data



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

Count Frequencies in training data

$$\hat{P}(c_j) =$$

$$\hat{P}(x_i | c_j) =$$



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

Count Frequencies in training data

$$\hat{P}(c_j) = \frac{N_{c_j}}{N_{total}}$$

$$\hat{P}(x_i | c_j) =$$

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

Count Frequencies in training data

$$\hat{P}(c_j) = \frac{N_{c_j}}{N_{total}}$$

$$\hat{P}(x_i | c_j) = \frac{\text{count}(x_i, c_i)}{\sum_{x \in V} \text{count}(x, c_i)}$$

fraction of times word x_i appears
among all words in documents of topic c_i

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

Count Frequencies in training data

$$\hat{P}(c_j) = \frac{N_{c_j}}{N_{total}}$$

Maximum Likelihood Estimation

$$\hat{P}(x_i | c_j) = \frac{\text{count}(x_i, c_i)}{\sum_{x \in V} \text{count}(x, c_i)}$$

fraction of times word x_i appears
among all words in documents of topic c_j

What if we have seen no training *positive* documents with the word *fantastic*?

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

Probability of class will be 0, regardless of other words

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

$$\begin{aligned}\hat{P}(x_i|c_j) &= \frac{\text{count}(x_i, c_i) + 1}{\sum_{x \in V} (\text{count}(x, c_i) + 1)} \\ &= \frac{\text{count}(x_i, c_i) + 1}{(\sum_{x \in V} \text{count}(x, c_i)) + |V|}\end{aligned}$$

Laplacian smoothing (add 1)

Learning a Naive Bayes Classifier



- From training corpus, extract *Vocabulary*

- Calculate $P(c_j)$ terms

- For each c_j in C do

$docs_j \leftarrow$ all docs with class = c_j

$$P(c_j) \leftarrow \frac{|docs_j|}{|\text{total \# documents}|}$$

- Calculate $P(w_k | c_j)$ terms

- $Text_j \leftarrow$ single doc containing all $docs_j$

- For each word w_k in *Vocabulary*

$n_k \leftarrow$ # of occurrences of w_k in $Text_j$

$$P(w_k | c_j) \leftarrow \frac{n_k + \alpha}{n + \alpha |Vocabulary|}$$

Predict with Naive Bayes Classifier



Give a document composed of words X
choose the class c
that maximizes the Naive Bayes equation

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



scikit-learn uses a standard set of functions for all models

The two main ones for our purposes

- `model.fit(X, y)` — train the model with the given data set
- `model.predict(X_test)` — get predictions for the given test set



Unknown Words

- words that are not in our training data but are in our test data
- **Ignore them**
 - Pretend they are not in our test

Stop Words

- For NB, removing them doesn't usually help



Naive Bayes Example

What label should we predict for test?



	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
Test	?	predictable with no fun

	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
Test	?	predictable with no fun

1. Prior from training:

$$\hat{P}(c_j) = \frac{N_{c_j}}{N_{total}}$$

2. Drop "with"

3. Likelihoods from training:

$$p(w_i|c) = \frac{\text{count}(w_i, c) + 1}{(\sum_{w \in V} \text{count}(w, c)) + |V|}$$

Hint: for this example, do we care about words not in test?

4. Scoring the test set: