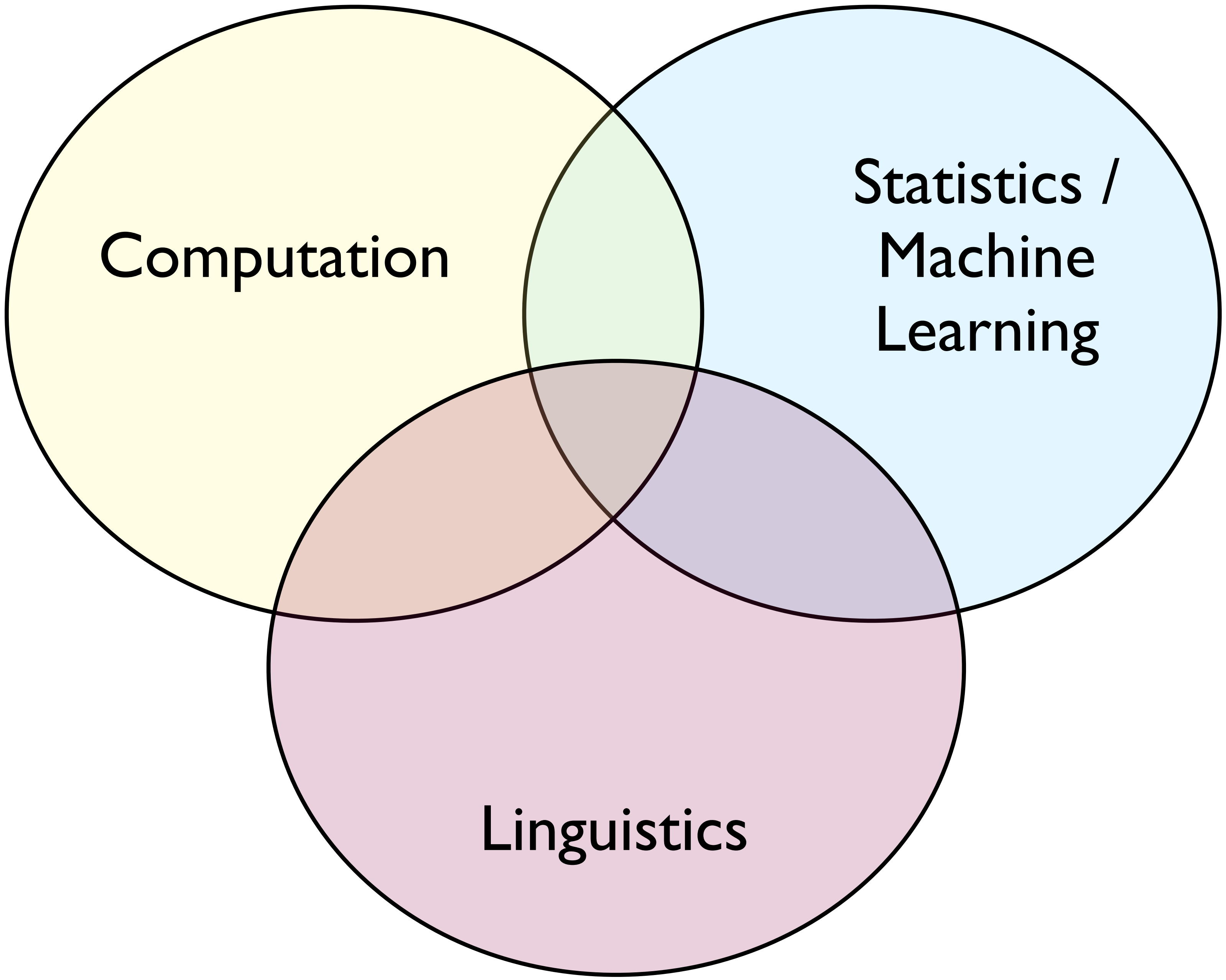
Lecture 6
Classification: Naive Bayes

Intro to NLP, CS585, Fall 2014

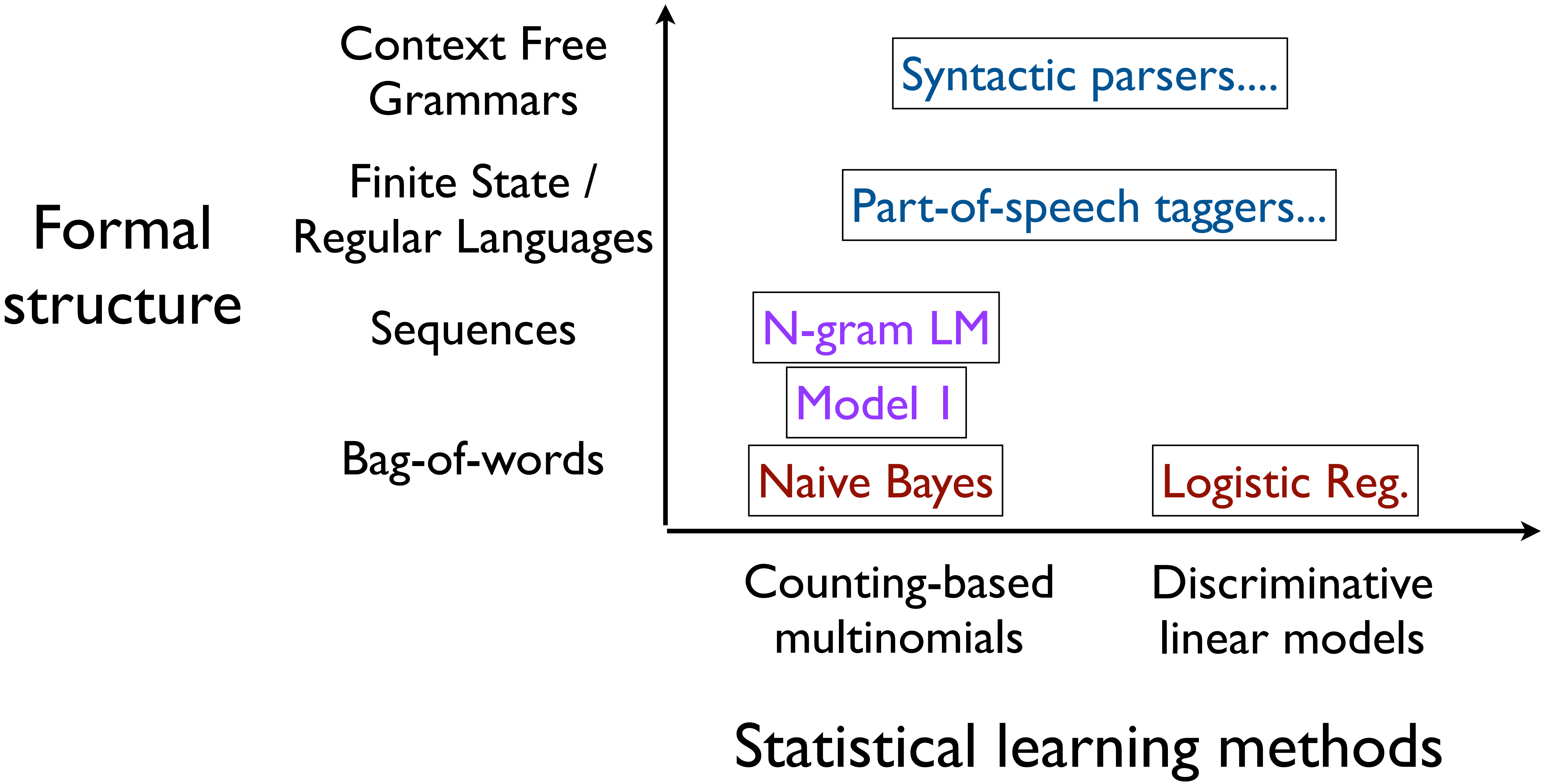
http://people.cs.umass.edu/~brenocon/inlp2014/

Brendan O'Connor (http://brenocon.com)

This course includes



Computation/Statistics in NLP (in this course)



Classification problems

- Given text ***d***, want to predict label ***y***
  - Is this restaurant review positive or negative?
  - Is this email spam or not?
  - Which author wrote this text?
  - (Is this word a noun or verb?)
- ***d***: documents, sentences, etc.
- ***y***: discrete/categorical variable

Goal: from training set of (d,y) pairs, *learn* a probabilistic classifier f(d) = P(y|d) ("supervised learning")

Features for model: 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
…	…

Figure 6.1 Intuition of the multinomial naive Bayes classifier applied to a movie review. The position of the words is ignored (the *bag of words* assumption) and we make use of the frequency of each word.

Generative vs. Discriminative approaches

Goal: from training set of (d,y) pairs, *learn* a probabilistic "classifier" f(d) = P(y|d)

Generative model: use the "noisy channel" idea.

$$P(y \mid d) \propto P(y)\ P(d \mid y; \theta)$$

Learning: $\max_\theta \prod_{i \in train} P(d_i \mid y_i; \theta)$ (where it's just counting)

Naive Bayes

Discriminative model: directly learn this function

$$P(y \mid d) = f(d; \theta)$$

Learning: $\max_\theta \prod_{i \in train} P(y_i \mid d_i)$ (where it's harder than counting)

Logistic Regression

Multinomial Naive Bayes: Unigram LM

Tokens in doc ↓

$$P(y \mid w_1..w_T) \propto P(y)\ P(w_1..w_T \mid y)$$

↓

conditional independence assumption $\prod_t P(w_t \mid y)$

- <u>Generative story:</u>
- Choose doc category **y**
- For each token position in doc:
  - Draw ***w_t***

Parameters: $P(w \mid y)$ for each document category **y** and wordtype ***w***

$P(y)$ prior distribution over document categories ***y***

<u>*Learning*</u>: with pseudocount smoothing,

$$P(w \mid y, \alpha) = \frac{\#(w \text{ occurrences in docs with label } y) + \alpha}{\#(\text{tokens total across docs with label } y) + V\alpha}$$

Multinomial Naive Bayes: Unigram LM

Prediction

Infer most likely class for new document

$$\arg\max_k \; P(y=k) \; \prod_t P(w_t \mid y=k)$$

Infer posterior probabilities for new document

$$P(y=k \mid w_1..w_T) = \frac{P(y=k) \; \prod_t P(w_t \mid y=k)}{\sum_{k'} P(y=k') \prod_t P(w_t \mid y=k')}$$

Example

	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 originality

Learning

Estimate prior

$$P(-) = \frac{3}{5} \qquad P(+) = \frac{2}{5}$$

Estimate word likelihoods with pseudocount=1

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

$$P(\text{"with"}|-) = \frac{0+1}{14+20} \qquad P(\text{"with"}|+) = \frac{0+1}{9+20}$$

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

$$P(\text{"originality"}|-) = \frac{0+1}{14+20} \qquad P(\text{"originality"}|+) = \frac{0+1}{9+20}$$

Prediction/Inference

$$P(S|-)P(-) = \frac{3}{5} \times \frac{2 \times 1 \times 2 \times 1}{34^4} = 1.8 \times 10^{-6}$$

$$P(S|+)P(+) = \frac{2}{5} \times \frac{1 \times 1 \times 1 \times 1}{29^4} = 5.7 \times 10^{-7}$$

NB as a Linear Model

Consider: ratio of posterior probs

$$\frac{P(+ \mid w_1..w_T)}{P(- \mid w_1..w_T)}$$

>1 then + more likely
<1 then - more likely

Odds form of Bayes Rule:

prior ratio *likelihood ratio*

$$= \frac{P(+)}{P(-)} \frac{P(w_1..w_T \mid +)}{P(w_1..w_T \mid -)} \quad \frac{1/P(w_1..w_T)}{1/P(w_1..w_T)}$$

$$= \frac{P(+)}{P(-)} \frac{\prod_t P(w_t|+)}{\prod_t P(w_t|-)}$$

$$= \frac{3}{5} \quad \frac{2/34 \times 1/34 \times 2/34 \times 1/34}{1/29 \times 1/29 \times 1/29 \times 1/29}$$

NB as a Linear Model

$$\frac{P(+ \mid w_1..w_T)}{P(- \mid w_1..w_T)} = \frac{P(+)}{P(-)} \frac{\prod_t P(w_t|+)}{\prod_t P(w_t|-)}$$

>1 then + more likely

<1 then - more likely

$$= \frac{P(+)}{P(-)} \prod_t \frac{P(w_t|+)}{P(w_t|-)}$$

$$\log \frac{P(+ \mid w_1..w_T)}{P(- \mid w_1..w_T)} = \log \frac{P(+)}{P(-)} + \sum_t \log \frac{P(w_t|+)}{P(w_t|-)}$$

>0 then + more likely

<0 then - more likely

$$= \log \frac{P(+)}{P(-)} + \sum_w^V n_w \log \frac{P(w|+)}{P(w|-)}$$

$$= \log \frac{3}{5} + \log \frac{2/34}{1/29} + \log \frac{1/34}{1/29} + \log \frac{2/34}{1/29} + \log \frac{1/34}{1/29}$$

NB as a Linear Model

$$\log \frac{P(+ \mid w_1..w_T)}{P(- \mid w_1..w_T)} = \log \frac{P(+)}{P(-)} + \sum_{w}^{V} n_w \underbrace{\log \frac{P(w|+)}{P(w|-)}}_{\beta_w}$$

>0 then + more likely
<0 then - more likely

$$= \underbrace{}_{\beta_0} + \underbrace{}_{(\beta_{1:V})^\mathsf{T}\mathbf{n}}$$

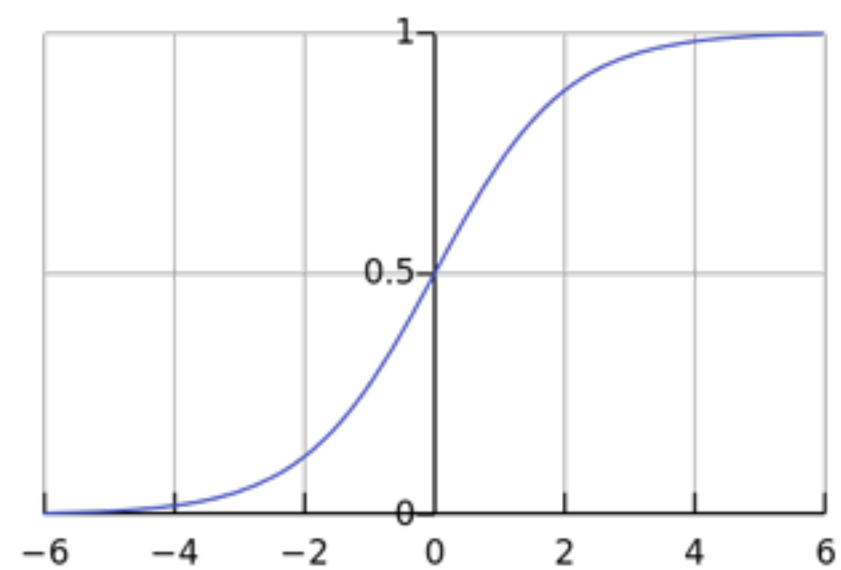
$$= \beta^\mathsf{T}\mathbf{x}$$

Where
$\boldsymbol{x}$ = (1, count "happy", count "sad", ...) ***Feature vector***

$$P(+ \mid w_1..w_T) = \frac{\exp(\beta^\mathsf{T}\mathbf{x})}{1 + \exp(\beta^\mathsf{T}\mathbf{x})}$$

Logistic sigmoid function

$$g(z) = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}}$$



Logistic regression

$$P(+ \mid w_1..w_T) = \frac{\exp(\beta^\mathsf{T}\mathbf{x})}{1 + \exp(\beta^\mathsf{T}\mathbf{x})}$$

- NB (decision between unigram LMs) prescribes one particular formula for the beta weights.
- Can we just fit the beta weights to maximize likelihood of the training data?