NUS National University of Singapore | School of Computing

Course Revision

CS4248 Natural Language Processing

Week 13

Min-Yen KAN

13

Slide Credit: David Bamman (UC Berkeley)

How do you go out and solve new problems involving text?

Slide Credit: David Bamman (UC Berkeley)

1. Language has structure

Slide Credit: David Bamman (UC Berkeley)

“… is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius”

Roger Ebert, *Apocalypse Now*

“I hated this movie. Hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it.”

Roger Ebert, *North*

Slide Credit: David Bamman (UC Berkeley)

Bag of Words

Representation of text only as the counts of words that it contains

	Apocalypse Now	North
the	1	1
of	0	0
hate	0	9
genius	1	0
bravest	1	0
stupid	0	1
like	0	1
…		

Slide Credit: David Bamman (UC Berkeley)

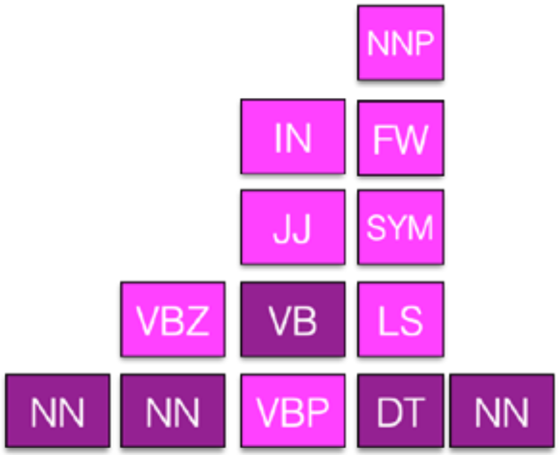
Language Model

"Hillary Clinton seemed to add Benghazi to her already-long list of culprits to blame for her upset loss to Donald ________"

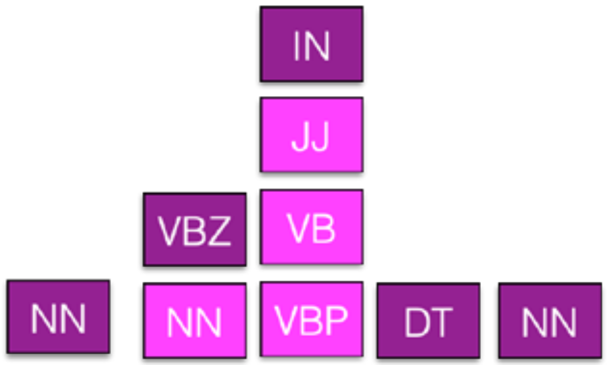
Slide Credit: David Bamman (UC Berkeley)
Source Text: http://www.foxnews.com/politics/2017/09/13/clinton-laments-how-benghazi-tragedy-hurt-her-politically.html

POS Tagging

Labeling the tag that is correct **for the context**.



Fruit flies like a banana.



Time flies like an arrow.

Slide Credit: David Bamman (UC Berkeley)

Word Senses

original	It urged that the city take steps to remedy this problem
lemma sense	It urge^{1} that the city^{2} take^{1} step^{1} to remedy^{1} this problem^{2}
synset number	It urge$^{2:32:00}$ that the city$^{1:15:01}$ take$^{2:41:04}$ step$^{1:04:02}$ to remedy$^{2:30:00}$ this problem$^{1:10:00}$

Slide Credit: David Bamman (UC Berkeley)

Thematic Roles

The thematic roles for verbs generally are predictable by the syntactic position of the argument (specific to each verb class). Some allow for consistent alternations:

Agent Theme Goal

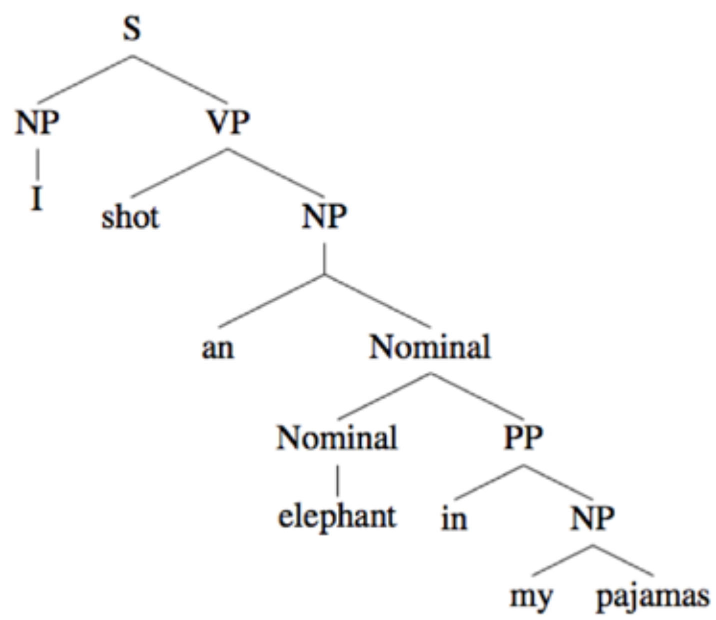
Doris gave the book to Cary

Agent Goal Theme

Doris gave Cary the book

Slide Credit: David Bamman (UC Berkeley)

Phrase Structure Syntax



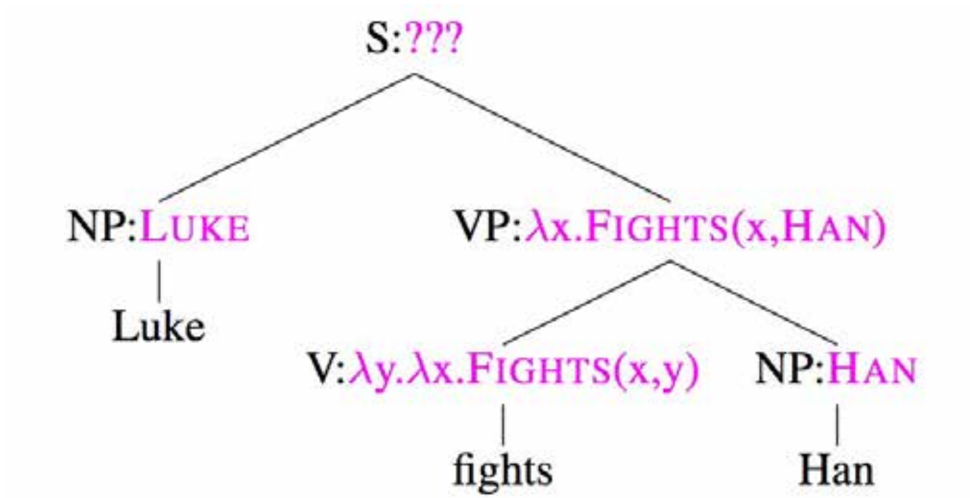
Every internal node is a phrase

my pajamas
in my pajamas
elephant in my pajamas
an elephant in my pajamas
shot an elephant in my pajamas
I shot an elephant in my pajamas

Each phrase could be replaced by another of the same type of constituent

Slide Credit: David Bamman (UC Berkeley)

Compositional Semantics



Slide Credit: David Bamman (UC Berkeley)

Coreference

LUKE
I'll never join you!

VADER
If you only knew the power of the dark side. Obi-Wan never told you what happened to your father.

LUKE
He told me enough! It was you who killed him.

VADER
No. I am your father.

LUKE
No. No. That's not true! That's impossible!

VADER
Search your feelings. You know it to be true.

LUKE
No! No! No!

Slide Credit: David Bamman (UC Berkeley)

2. Most new problems can be solved with a familiar class of algorithms

Slide Credit: David Bamman (UC Berkeley)

- Classification
- Sequence Labeling
- Trees
- Graphs

Counting and normalizing (NB, PCFG, HMM)

Loglinear (logistic regression, MEMM, CRF)

Neural (CNN, RNN, LSTM, seq2seq, attention)

Slide Credit: David Bamman (UC Berkeley)

Classification

Slide Credit: David Bamman (UC Berkeley)

Bayes' Rule

Likelihood: How probable is the data given that our document is a member of y?

Prior: How probable is a document to be a member of class y seeing any data?

$$P(y|\boldsymbol{w}) = \frac{P(\boldsymbol{w}|y)P(y)}{P(\boldsymbol{w})}$$

Posterior: How probable is the instance classified as a member of class y?

Marginal: How probable is the evidence under any class?

Slide adapted from CS3244 Machine Learning

Naïve Bayes Classifier

Training a Naïve Bayes classifier consists of estimating these two quantities from training data for all classes Y

At test time, use those estimated probabilities to calculate the posterior probability of each class y and select the class with the highest probability

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

Maximum a posteriori or mostly likely class

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

Bayes rule

$$= \underset{c \in \mathcal{C}}{\operatorname{argmax}}\, P(d|c)P(c)$$

Dropping the P(d) in the denominator

$$= \underset{c \in \mathcal{C}}{\operatorname{argmax}}\, \overbrace{P(f_1, f_2, ..., f_n|c)}^{\text{likelihood}}\,\overbrace{P(c)}^{\text{prior}}$$

Document d represented as features $f_1, ..., f_n$ (such as word counts) BoW assumption

$$= \underset{c \in \mathcal{C}}{\operatorname{argmax}}\, P(f_1|c)P(f_2|c)...P(f_n)|c)P(c)$$

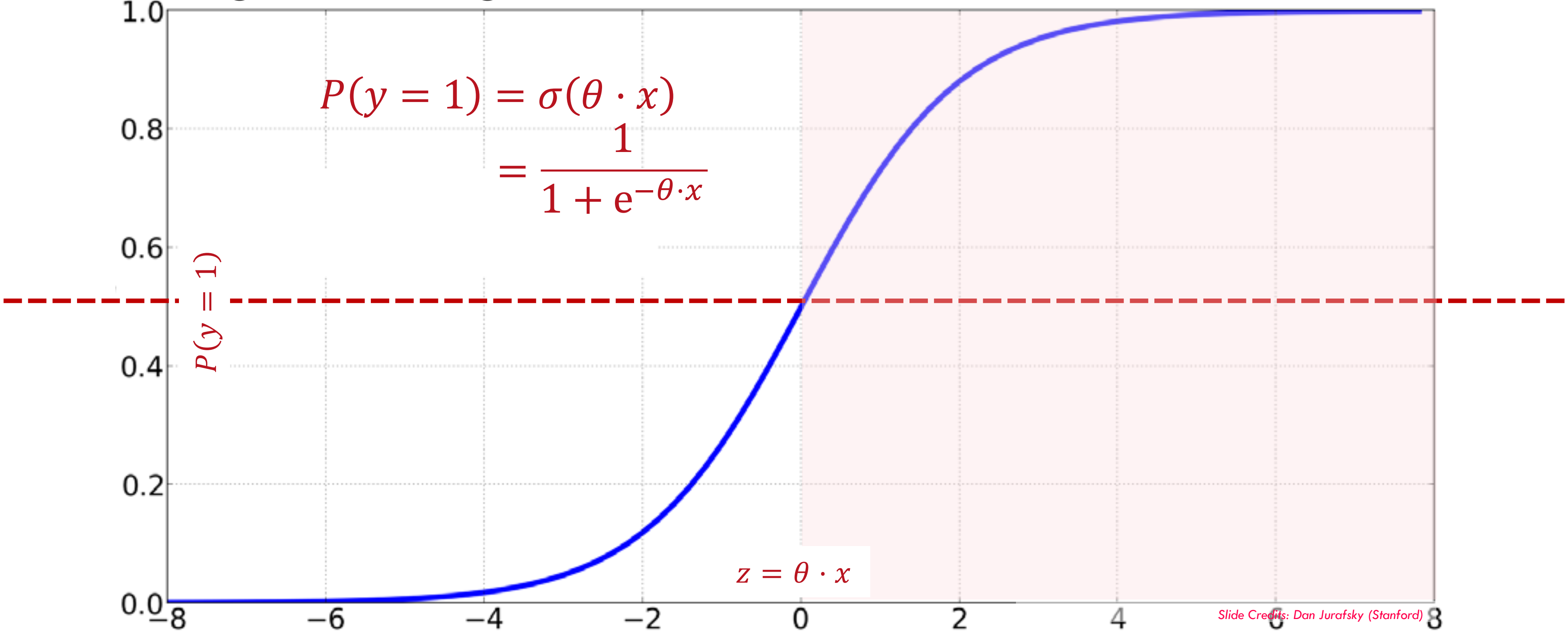
Independence Assumption

$$c_{NB} = \underset{c \in \mathcal{C}}{\operatorname{argmax}}\, \boxed{P(c)} \prod_{f \in \mathcal{F}} \boxed{P(f|c)}$$

Equation for NB classifier

Slide Credits: David Bamman (UCB)

Logistic Regression



Slide Credits: Dan Jurafsky (Stanford)

X = feature vector

Feature	Value
the	0
and	0
bravest	0
love	0
loved	0
genius	0
not	0
fruit	1
BIAS	1

W = coefficients

Feature	w
the	0.01
and	0.03
bravest	1.4
love	3.1
loved	1.2
genius	0.5
not	-3.0
fruit	-0.8
BIAS	-0.1

Slide Credit: David Bamman (UC Berkeley)

Features

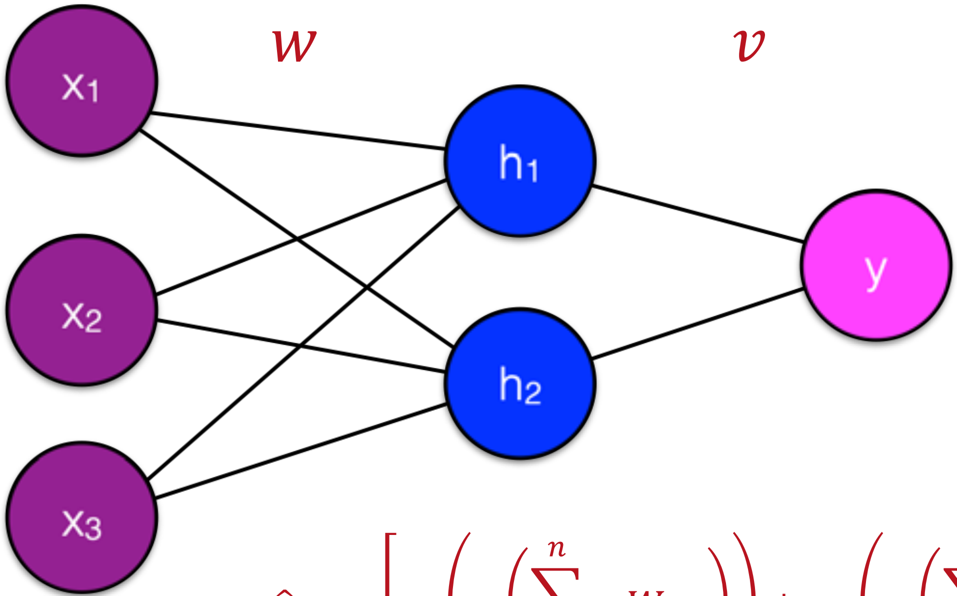
As a discriminative classifier, logistic regression doesn't assume features are independent like Naive Bayes does.

Its power partly comes in the ability to create richly expressive features with out the burden of independence.

We can represent text through features that are not just the identities of individual words, but any feature that is scoped over the entirety of the input.

features
contains *like*
has word that shows up in positive sentiment dictionary
review begins with "*I like*"
at least 5 mentions of positive affectual verbs (*like*, *love*, etc.)

Slide Credit: David Bamman (UC Berkeley)



$$\hat{y} = \sigma\left[v_1\left(\sigma\left(\sum_i^n x_i W_{i,1}\right)\right) + v_2\left(\sigma\left(\sum_i^n x_i W_{i,2}\right)\right)\right]$$

We can express y as a function only of the input x and the weights W and V

Slide Credit: David Bamman (UC Berkeley)

Sequences

Slide Credit: David Bamman (UC Berkeley)

Sequence Labeling

Sequence labeling problems make a labeling decision at each timestep

B-PER	I-PER	O	O	O	O	B-ORG
Tim	Cook	is	the	CEO	of	Apple

0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
#	b	l	a	c	k	l	i	v	e	s	m	a	t	t	e	r

Slide Credit: David Bamman (UC Berkeley)

Sequence Labeling

School of Computing

$$X = \{x_1, \dots, x_n\}$$

$$Y = \{y_1, \dots, y_n\}$$

For a set of inputs x with n sequential time steps, one corresponding label y_i for each x_i

Model the structure that exists between within y

Slide Credit: David Bamman (UC Berkeley)

HMM

$$P(x_1, \ldots, x_n, y_1, \ldots, y_n) \approx \prod_{i=1}^{n+1} P(y_i \mid y_{i-1}) \prod_{i=1}^{n} P(x_i \mid y_i)$$

Slide Credit: David Bamman (UC Berkeley)

Hidden Markov Model

$$P(x \mid y) = P(x_1, \ldots, x_n \mid y_1, \ldots, y_n)$$

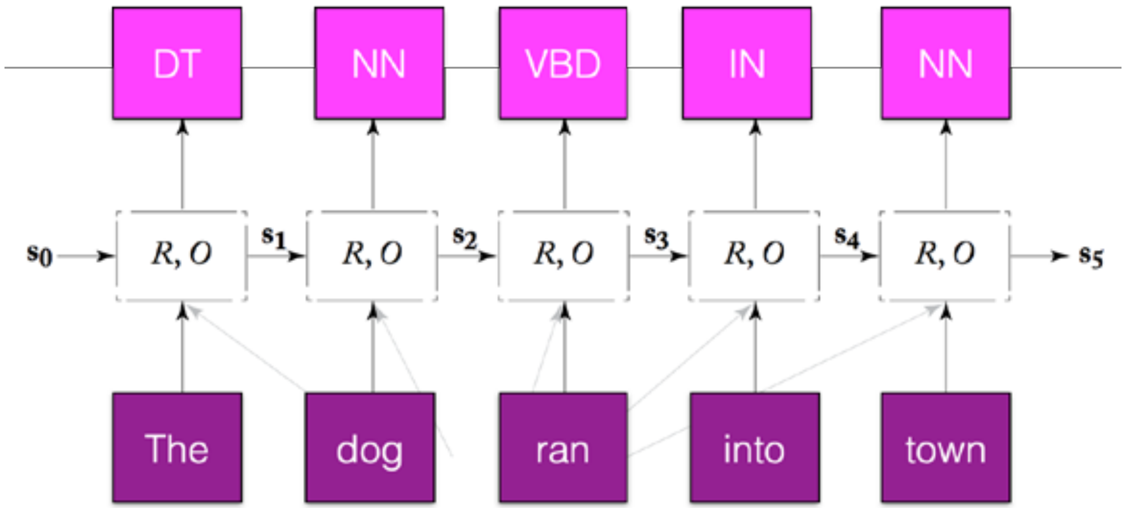
$$P(x_1, \ldots, x_n \mid y_1, \ldots, y_n) \approx \prod_{i=1}^{N} P(x_i \mid y_i)$$

Here again we'll make a strong assumption: the probability of the word we see at a given time step is only dependent on its label

Slide Credit: David Bamman (UC Berkeley)

Recurrent Neural Network

Predict the tag conditioned on the context



Slide Credit: David Bamman (UC Berkeley)

Bidirectional RNN

A powerful alternative is make predictions conditioning both on the past and the future.

Two RNNs

- One running left-to-right
- One right-to-left

Each produces an output vector at each time step, which we concatenate

Slide Credit: David Bamman (UC Berkeley)

Trees

Slide Credit: David Bamman (UC Berkeley)

PCFG

Probabilistic context-free grammar: each production is also associated with a probability.

This lets us calculate the probability of a parse for a given sentence; for a given parse tree T for sentence S comprised of n rules from R (each $A \rightarrow \beta$):

$$P(T, S) = \prod_{i}^{n} P(\beta | A)$$

Slide Credit: David Bamman (UC Berkeley)

Estimating PCFGs

$$\sum_{\beta} P(\beta|A) = \frac{Count(A \rightarrow \beta)}{\sum_i Count(A \rightarrow i)}$$

Or equivalently,

$$\sum_{\beta} P(\beta|A) = \frac{Count(A \rightarrow \beta)}{Count(A)}$$

Slide Credit: David Bamman (UC Berkeley)

A		β	P(β | NP)
NP	→	NP PP	0.092
NP	→	DT NN	0.087
NP	→	NN	0.047
NP	→	NNS	0.042
NP	→	DT JJ NN	0.035
NP	→	NNP	0.034
NP	→	NNP NNP	0.029
NP	→	JJ NNS	0.027
NP	→	QP -NONE-	0.018
NP	→	NP SBAR	0.017
NP	→	NP PP-LOC	0.017
NP	→	JJ NN	0.015
NP	→	DT NNS	0.014
NP	→	CD	0.014
NP	→	NN NNS	0.013
NP	→	DT NN NN	0.013
NP	→	NP CC NP	0.013

Slide Credit: David Bamman (UC Berkeley)

Natural Language Generation

Slide Credit: David Bamman (UC Berkeley)

Language Model

Language modeling is the task of estimating $P(w)$

- Count and normalize
- Featurized
- Neural (RNN)

Slide Credit: David Bamman (UC Berkeley)

Encoder–Decoder Framework

Language modeling: predict a word given its left context.

How about when there's some prior information?

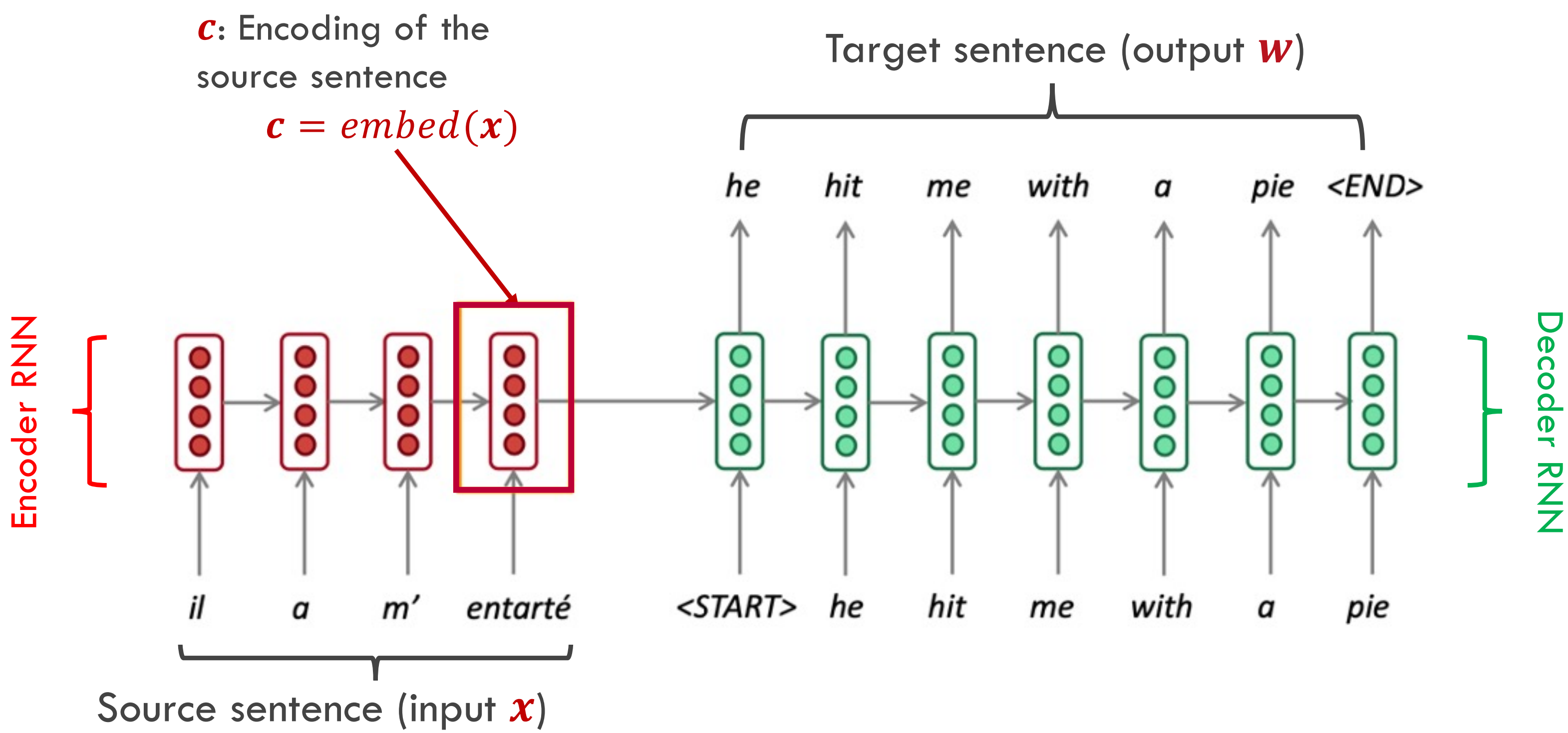
Conditional Language Model

- Question Answering: predict an answer given its left context and the source passage.
- Machine translation: predict a word given its left context and the full text of the source.

Basic idea: encode some context into a fixed vector; and then decode a new sentence from that embedding.

Slide Credit: David Bamman (UC Berkeley)

Encoder–Decoder

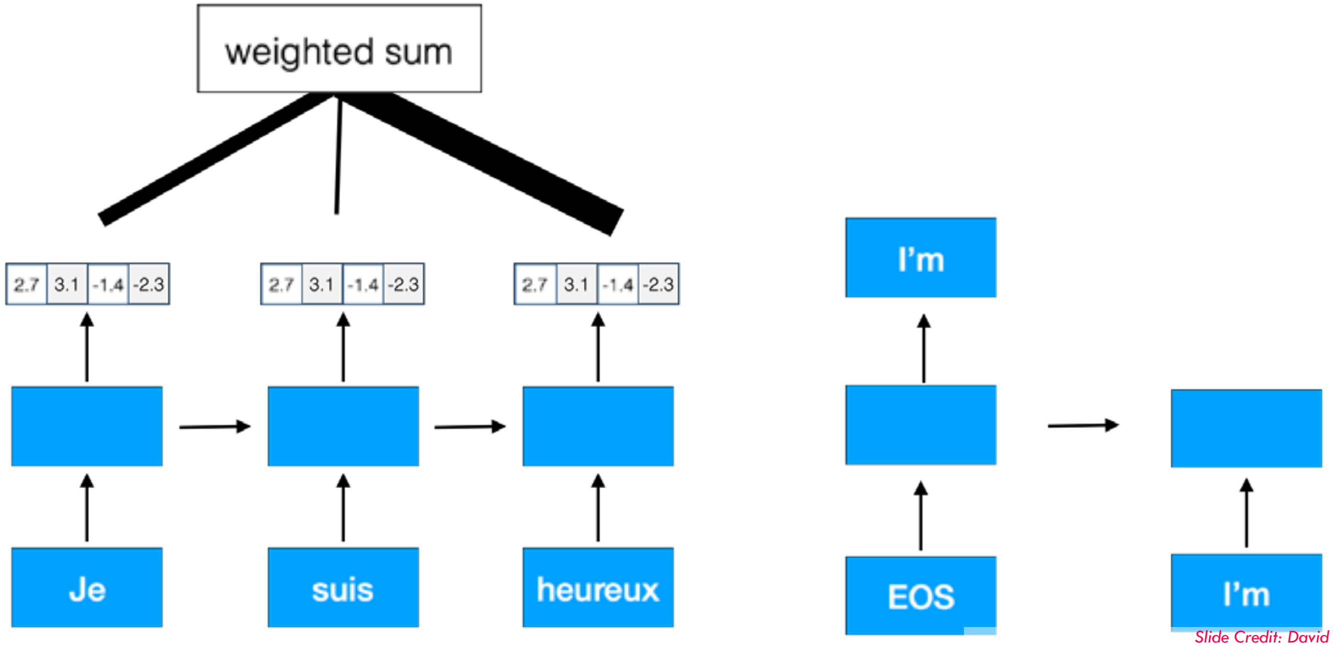


Adaptedd from Chris Manning (Stanford) CS224N

Encoder–Decoder with Attention

$$c = h_1 a_1 + h_2 a_2 + h_3 a_3$$

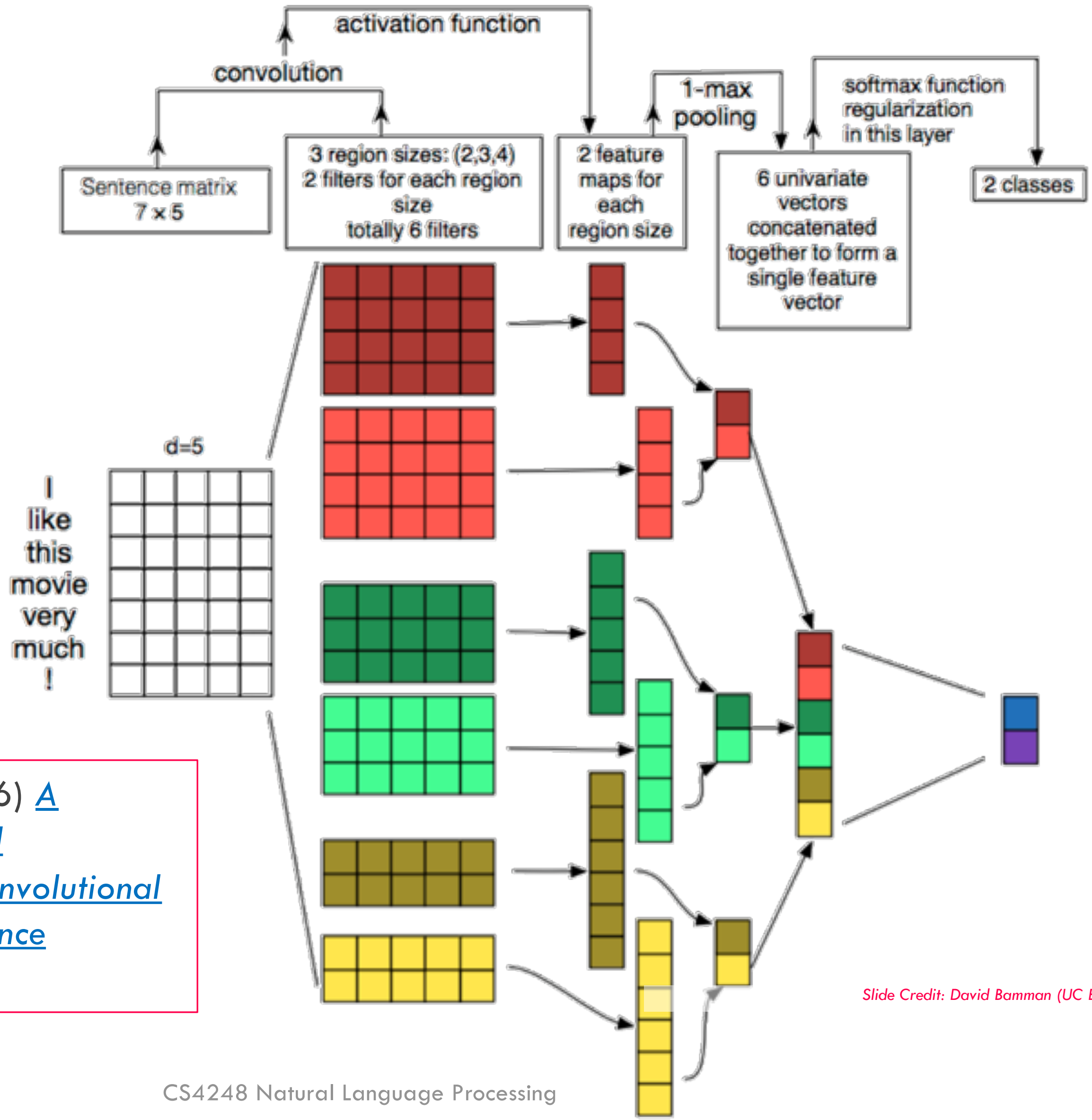
$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$



Slide Credit: David Bamman (UC Berkeley)

3. Neural methods are generally* better.

Slide Credit: David Bamman (UC Berkeley)

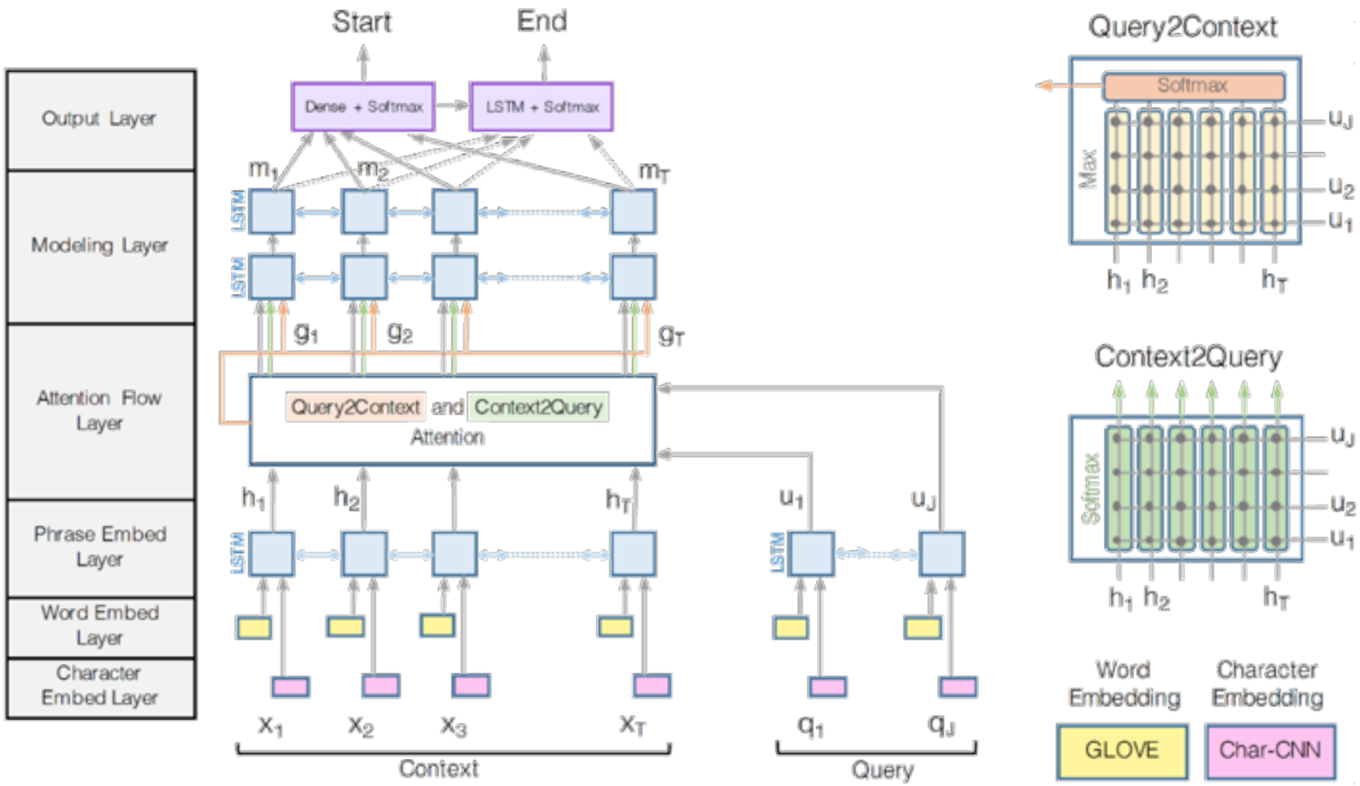


Zhang and Wallace (2016) [A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification]

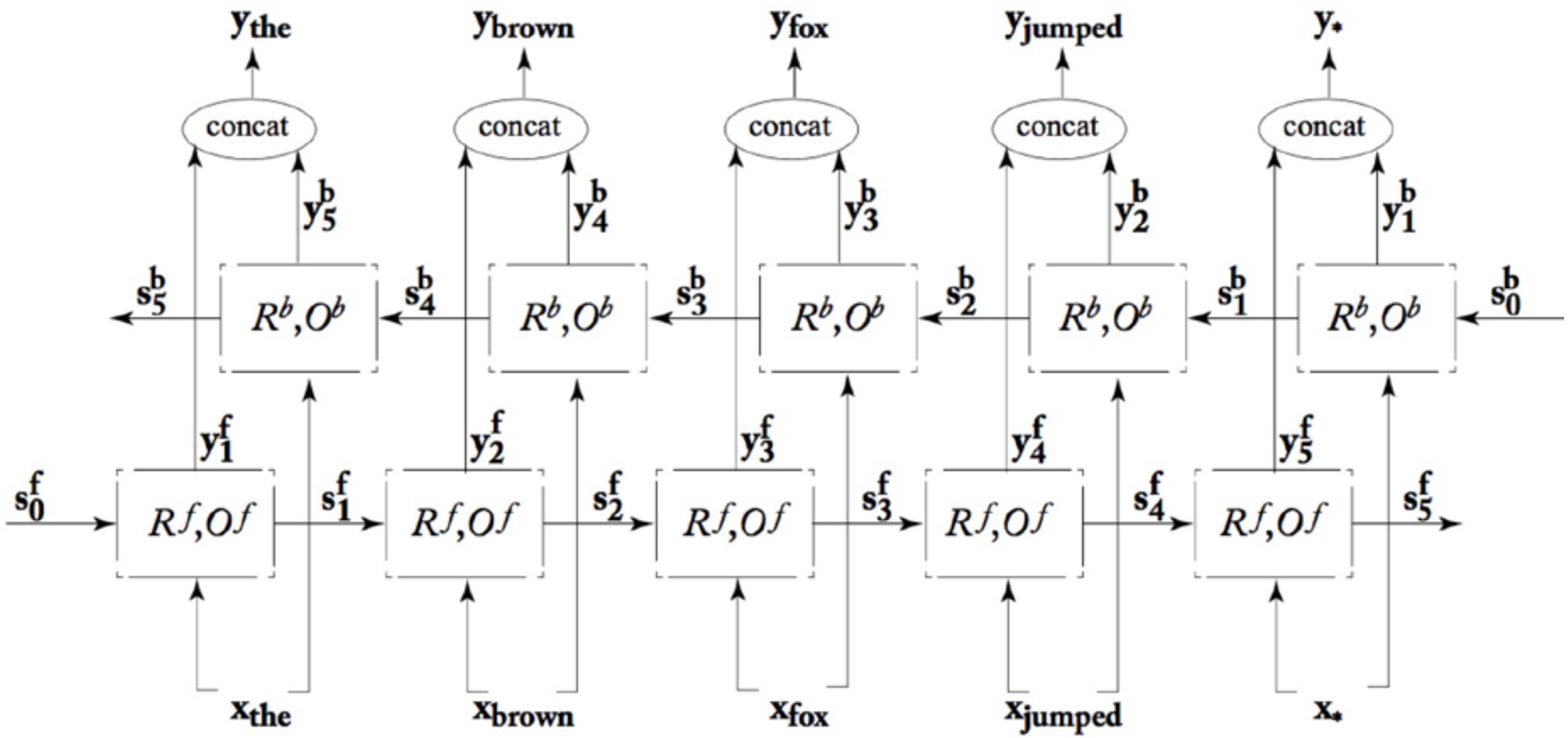
Slide Credit: David Bamman (UC Berkeley)

BiDAF

(Seo et al., 2017): *Bidirectional Attention Flow for Machine Comprehension*



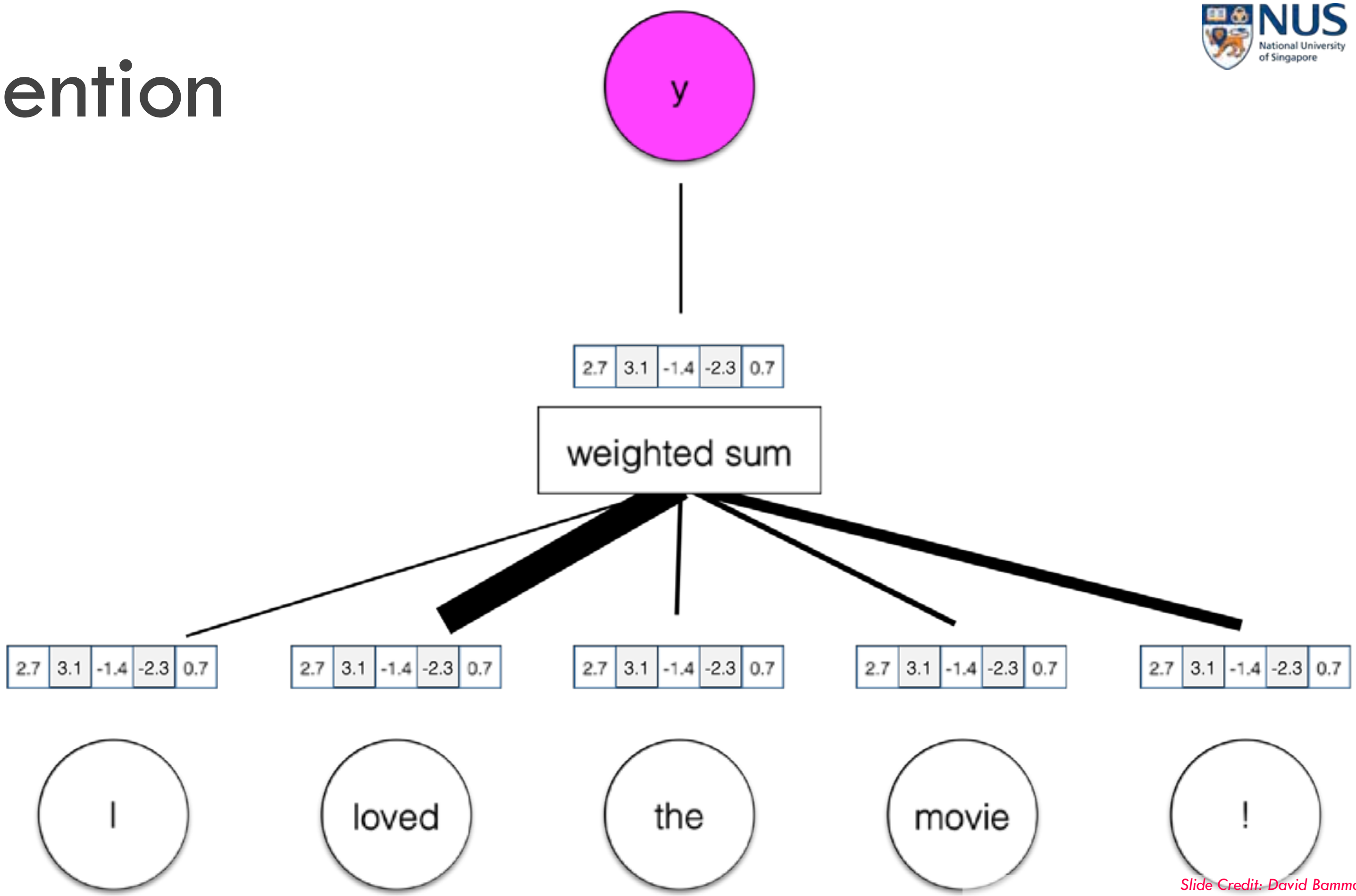
Slide Credits: Diyi Yang (Georgia Tech)



Goldberg (2017)

Slide Credit: David Bamman (UC Berkeley)

Attention

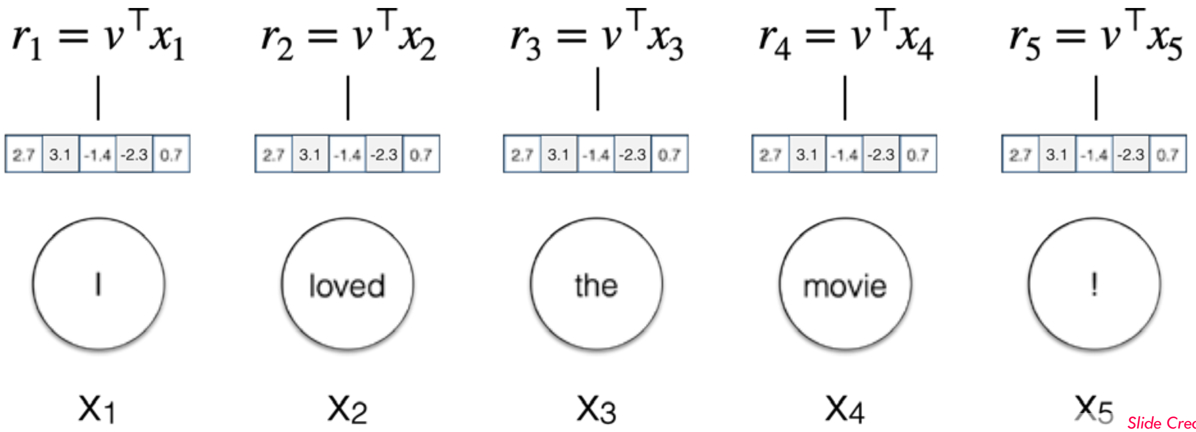


Slide Credit: David Bamman (UC Berkeley)

$v \in \mathscr{R}^H$

2.7	3.1	-1.4	-2.3	0.7

Define v to be a vector to be learned; think of it as an "important word" vector. The dot product here measures how similar each input vector is to that "important word" vector.



Slide Credit: David Bamman (UC Berkeley)

Lexical semantics

"You shall know a word by the company it keeps"

[Firth 1957]

Slide Credit: David Bamman (UC Berkeley)

Distributed Representation

Vector representation that encodes information about the distribution of contexts a word appears in

Words that appear in similar contexts have similar representations (and similar meanings, by the distributional hypothesis).

Slide Credit: David Bamman (UC Berkeley)

4. Evaluation is critical.

Slide Credit: David Bamman (UC Berkeley)

Interannotator Agreement

		Annotator 1	
		puppy	fried chicken
Annotator 2	puppy	6	3
	fried chicken	2	5

observed agreement = 11/16 = 68.75%

Slide Credit: David Bamman (UC Berkeley)
Source Image: https://twitter.com/teenybiscuit/status/705232709220769792/photo/1

Experiment Design

	training	development	testing
size	80%	10%	10%
purpose	training models	model selection	evaluation; never look at it until the very end

Slide Credit: David Bamman (UC Berkeley)

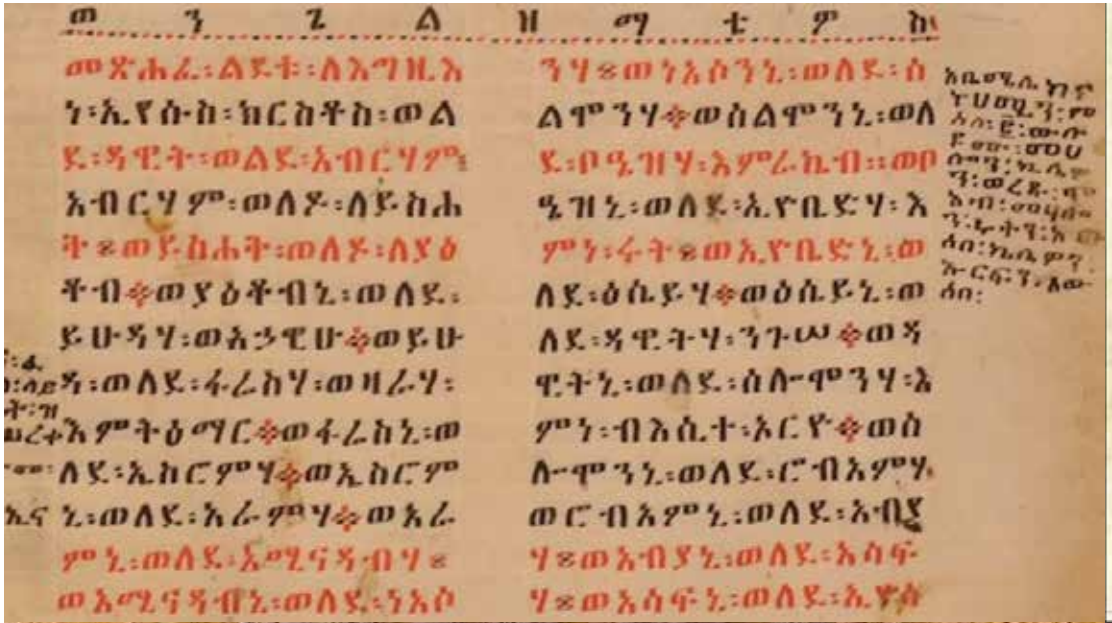
Metrics

- Perplexity
- Accuracy
- Precision/Recall/F_1
- Parseval (P/R/F_1 over labeled constituents)
- Correlation with human judgments
- BLEU Precision / ROUGE Recall

Slide Credit: David Bamman (UC Berkeley)

5. Text is data.

Slide Credit: David Bamman (UC Berkeley)

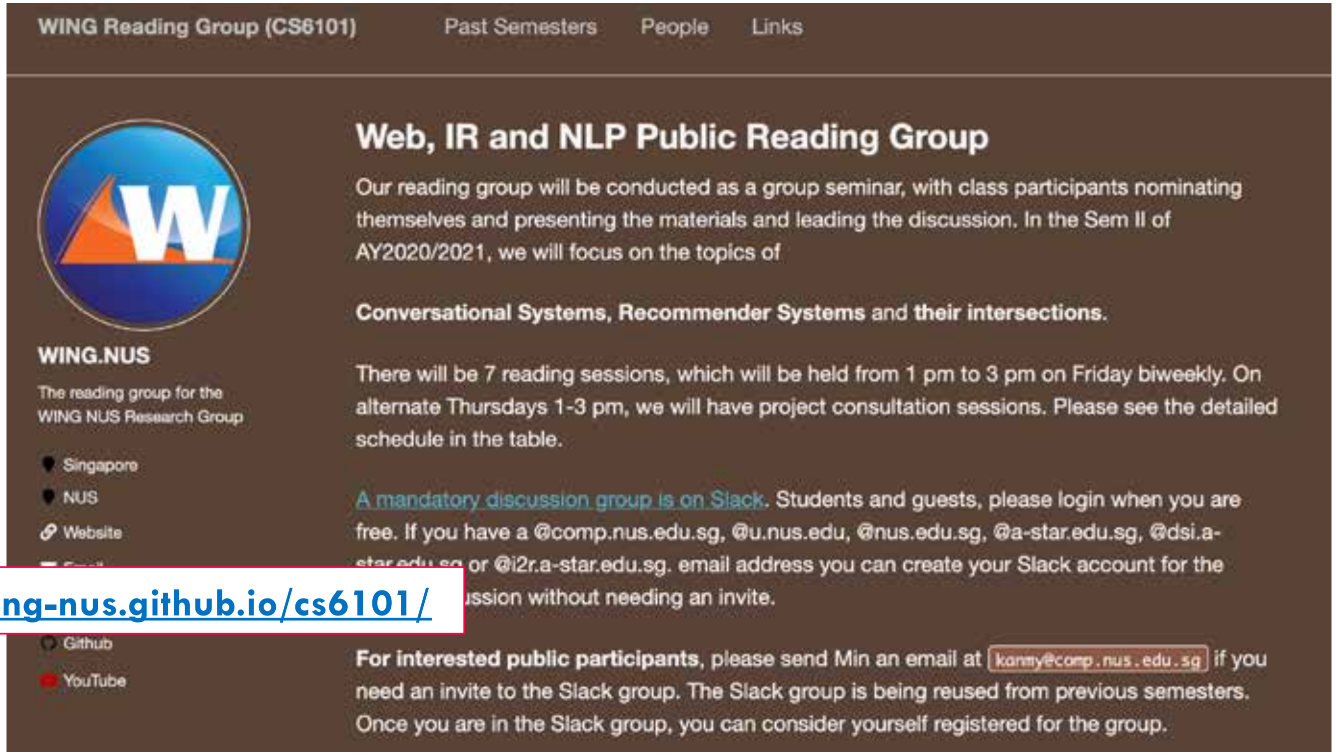



Slide Credit: David Bamman (UC Berkeley)

Keep in touch with CS4248!

https://www.facebook.com/cs4248

Level up with us!



https://wing-nus.github.io/cs6101/

NUS National University of Singapore | School of Computing

Goodbye!