

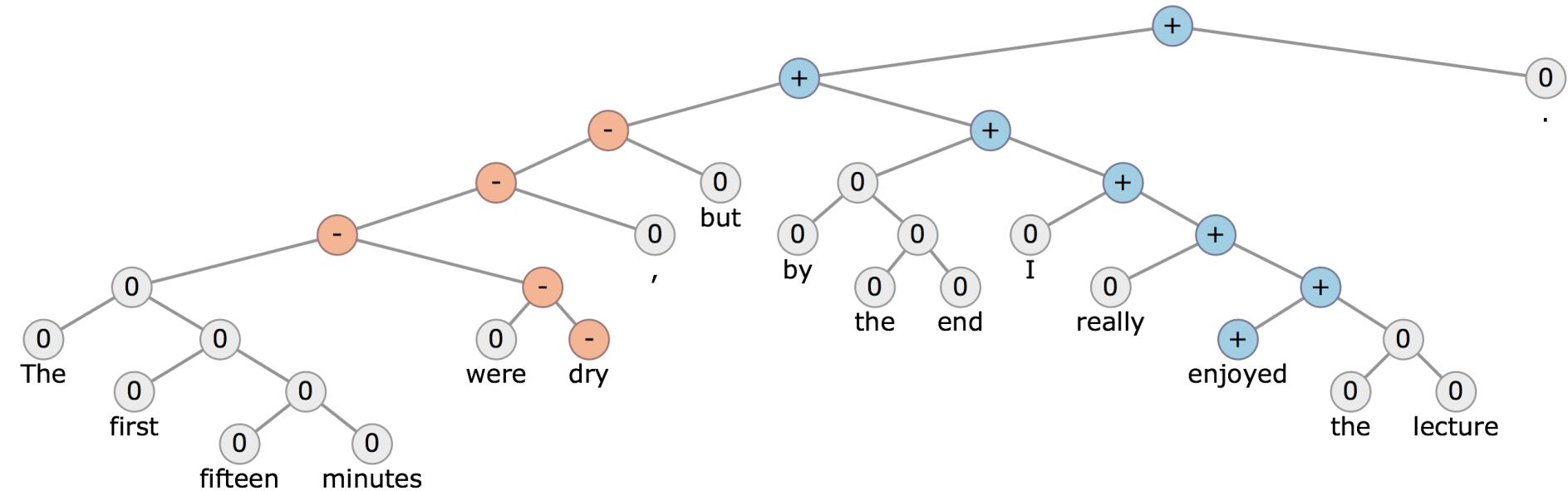
# **CS224d**

# **Deep Learning**

# **for Natural Language Processing**

**Richard Socher, PhD**

# Welcome



1. CS224d logistics
2. Introduction to NLP, deep learning and their intersection

# Course Logistics

- Instructor: Richard Socher  
(Stanford PhD, 2014; now Founder/CEO at MetaMind)
- TAs: James Hong, Bharath Ramsundar, Sameep Bagadia, David Dindi, ++
- Time: Tuesday, Thursday 3:00-4:20
- Location: Gates B1
- There will be 3 problem sets (with lots of programming),  
a midterm and a final project
- For syllabus and office hours, see <http://cs224d.stanford.edu/>
- Slides uploaded before each lecture, video + lecture notes after

# Pre-requisites

- Proficiency in Python
  - All class assignments will be in Python. There is a tutorial [here](#)
- College Calculus, Linear Algebra (e.g. MATH 19 or 41, MATH 51)
- Basic Probability and Statistics (e.g. CS 109 or other stats course)
- Equivalent knowledge of CS229 (Machine Learning)
  - cost functions,
  - taking simple derivatives
  - performing optimization with gradient descent.

# Grading Policy

- 3 Problem Sets:  $15\% \times 3 = 45\%$
- Midterm Exam: 15%
- Final Course Project: 40%
  - Milestone: 5% (2% bonus if you have your data and ran an experiment!)
  - Attend at least 1 project advice office hour: 2%
  - Final write-up, project and presentation: 33%
  - Bonus points for exceptional poster presentation
- Late policy
  - 7 free late days – use as you please
  - Afterwards, 25% off per day late
  - PSets Not accepted after 3 late days per PSet
  - Does not apply to Final Course Project
- Collaboration policy: Read the student code book and Honor Code!
- Understand what is ‘collaboration’ and what is ‘academic infraction’

# High Level Plan for Problem Sets

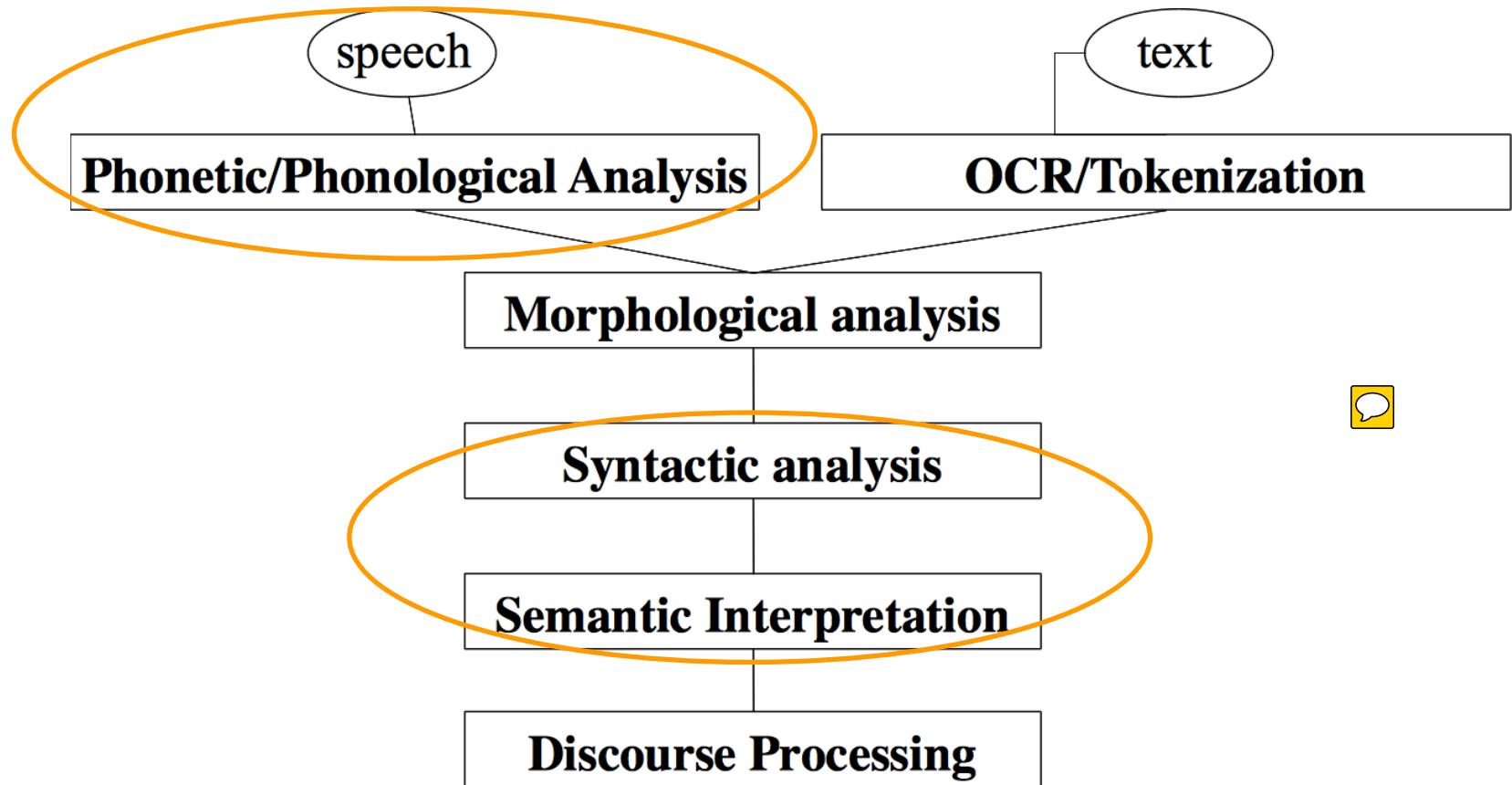
- The first half of the course and the first 2 PSets will be hard
- PSet 1 is in pure python code (numpy etc.) to really understand the basics
- Released on April 4th
- New: PSets 2 & 3 will be in TensorFlow, a library for putting together new neural network models quickly (→ special lecture)
- PSet 3 will be shorter to increase time for final project
- Libraries like TensorFlow (or Torch) are becoming standard tools
- But still some problems

# What is Natural Language Processing (NLP)?

- Natural language processing is a field at the intersection of
  - computer science
  - artificial intelligence
  - and linguistics.
- Goal: for computers to process or “understand” natural language in order to perform tasks that are useful, e.g.
  - Question Answering
- Fully **understanding and representing** the **meaning** of language (or even defining it) is an illusive goal.
- Perfect language understanding is AI-complete



# NLP Levels



# (A tiny sample of) NLP Applications

- Applications range from simple to complex:
- Spell checking, keyword search, finding synonyms
- Extracting information from websites such as
  - product price, dates, location, people or company names
- Classifying, reading level of school texts, positive/negative sentiment of longer documents
- Machine translation
- Spoken dialog systems
- Complex question answering

# NLP in Industry

- Search (written and spoken)
- Online advertisement
- Automated/assisted translation
- Sentiment analysis for marketing or finance/trading
- Speech recognition
- Automating customer support



3/18/11 at 4:00 PM | 17 Comments  
**Mentions of the Name ‘Anne Hathaway’ May Drive Berkshire Hathaway Stock**  
By Patrick Huguenin



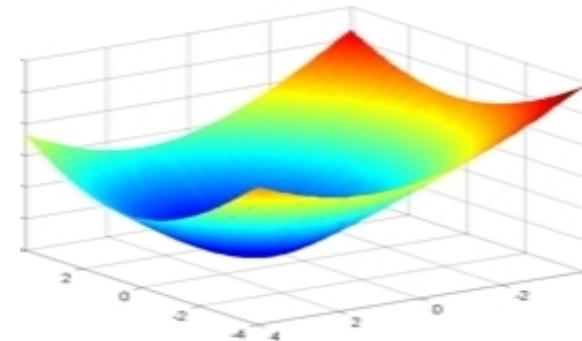
# Why is NLP hard?

- Complexity in representing, learning and using linguistic/  
situational/world/visual knowledge
- Jane hit June and then **she** [fell/run]. 
- Ambiguity: “I made her duck”

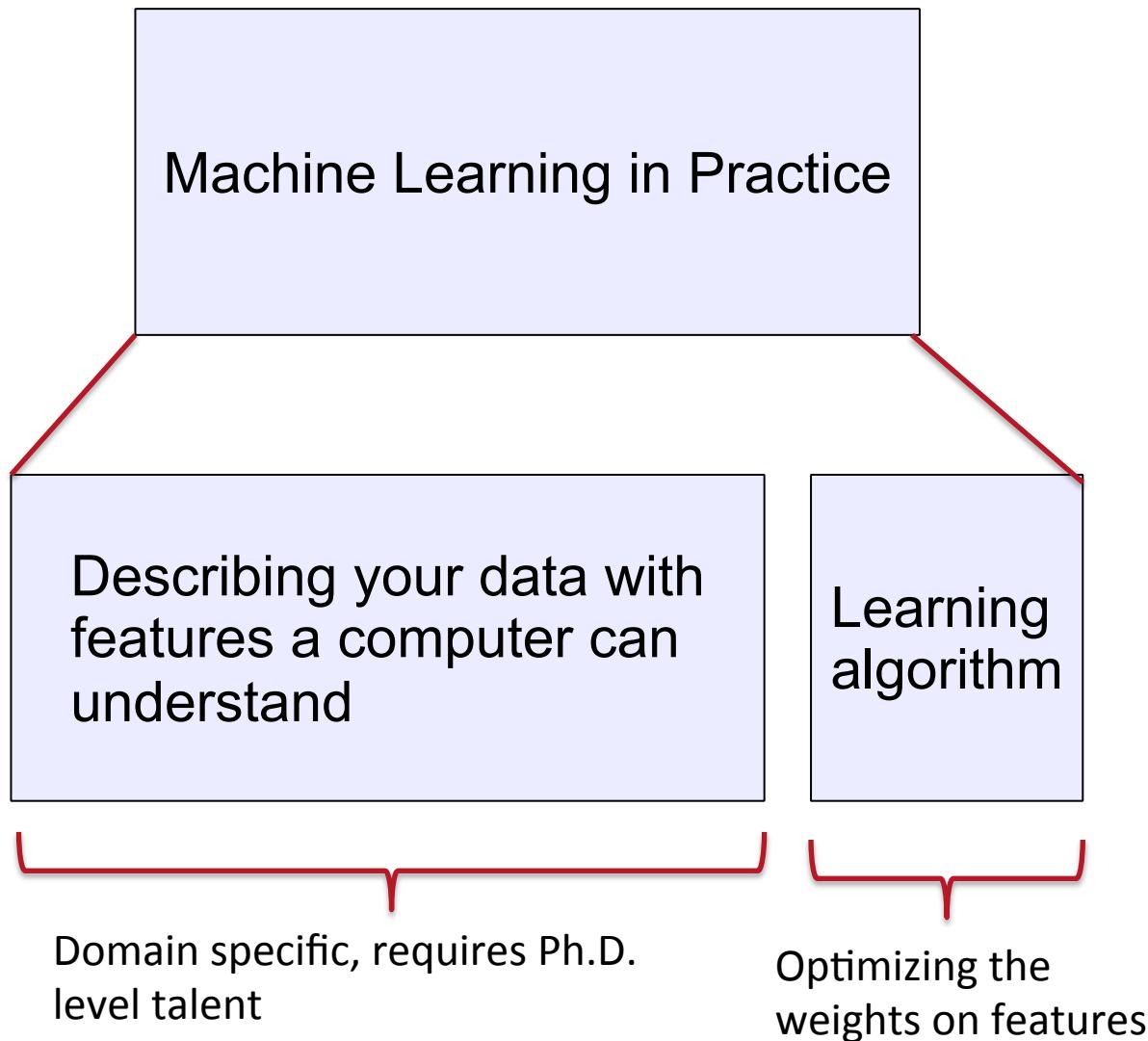
# What's Deep Learning (DL)?

- Deep learning is a subfield of machine learning
- Most machine learning methods work well because of human-designed representations and input features
  - For example: features for finding named entities like locations or organization names (Finkel, 2010):
- Machine learning becomes just optimizing weights to best make a final prediction

Feature	NER
Current Word	✓
Previous Word	✓
Next Word	✓
Current Word Character n-gram	all
Current POS Tag	✓
Surrounding POS Tag Sequence	✓
Current Word Shape	✓
Surrounding Word Shape Sequence	✓
Presence of Word in Left Window	size 4
Presence of Word in Right Window	size 4

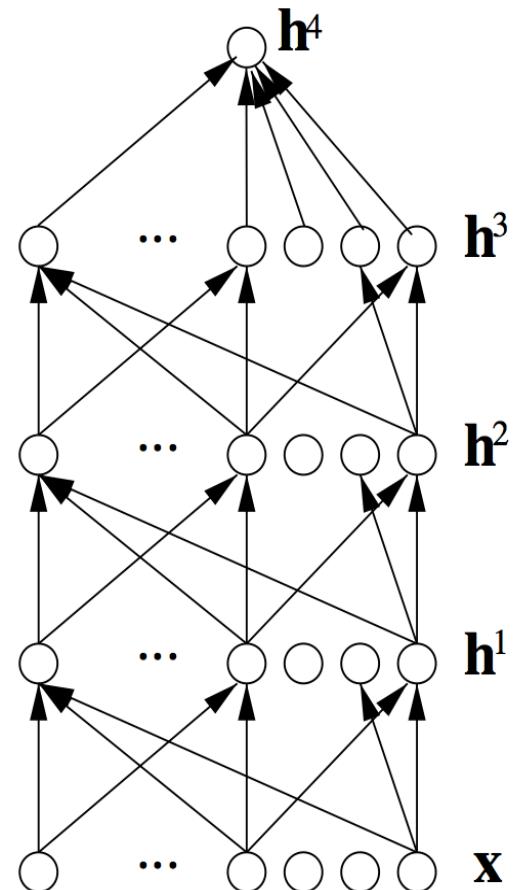


# Machine Learning vs Deep Learning



# What's Deep Learning (DL)?

- Representation learning attempts to automatically learn good features or representations
- Deep learning algorithms attempt to learn (multiple levels of) representation and an output
- From “raw” inputs  $\mathbf{x}$  (e.g. words)



# On the history and term of “Deep Learning”

- We will focus on different kinds of **neural networks**
- The dominant model family inside deep learning
- Only clever terminology for stacked logistic regression units?
  - Somewhat, but interesting modeling principles (end-to-end) and actual connections to neuroscience in some cases
- We will not take a historical approach but instead focus on methods which work well on NLP problems now
- For history of deep learning models (starting ~1960s), see:  
[Deep Learning in Neural Networks: An Overview](#)  
by Schmidhuber

# Reasons for Exploring Deep Learning

- Manually designed features are often over-specified, incomplete and take a long time to design and validate
- **Learned Features** are easy to adapt, fast to learn
- Deep learning provides a very flexible, (almost?) universal, learnable framework for **representing** world, visual and linguistic information.
- Deep learning can learn **unsupervised** (from raw text) and **supervised** (with specific labels like positive/negative)

# Reasons for Exploring Deep Learning

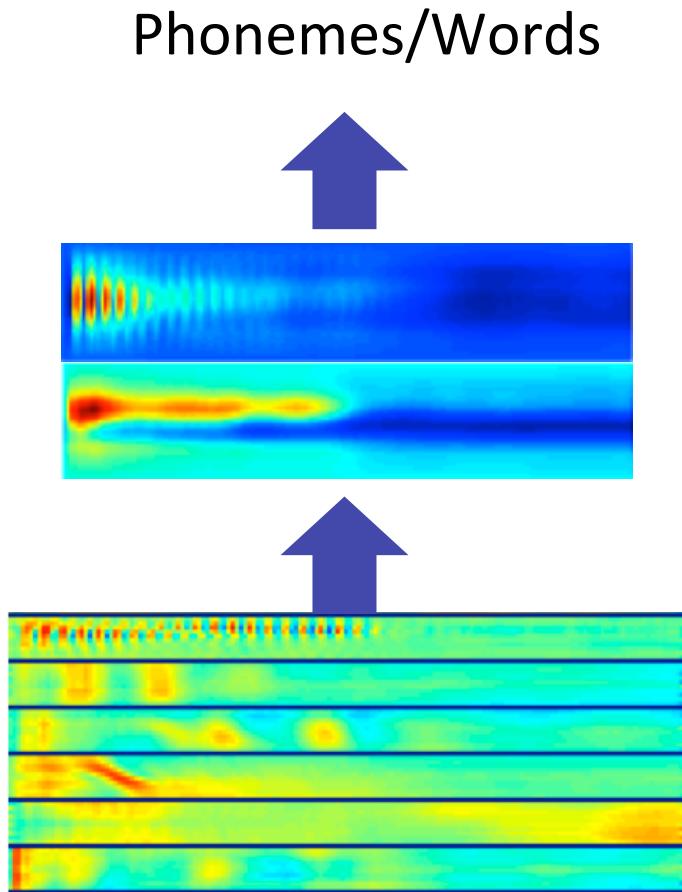
- In 2006 **deep** learning techniques started outperforming other machine learning techniques. Why now?
- DL techniques benefit more from a lot of data
- Faster machines and multicore CPU/GPU help DL
- New models, algorithms, ideas

→ **Improved performance** (first in speech and vision, then NLP)

# Deep Learning for Speech

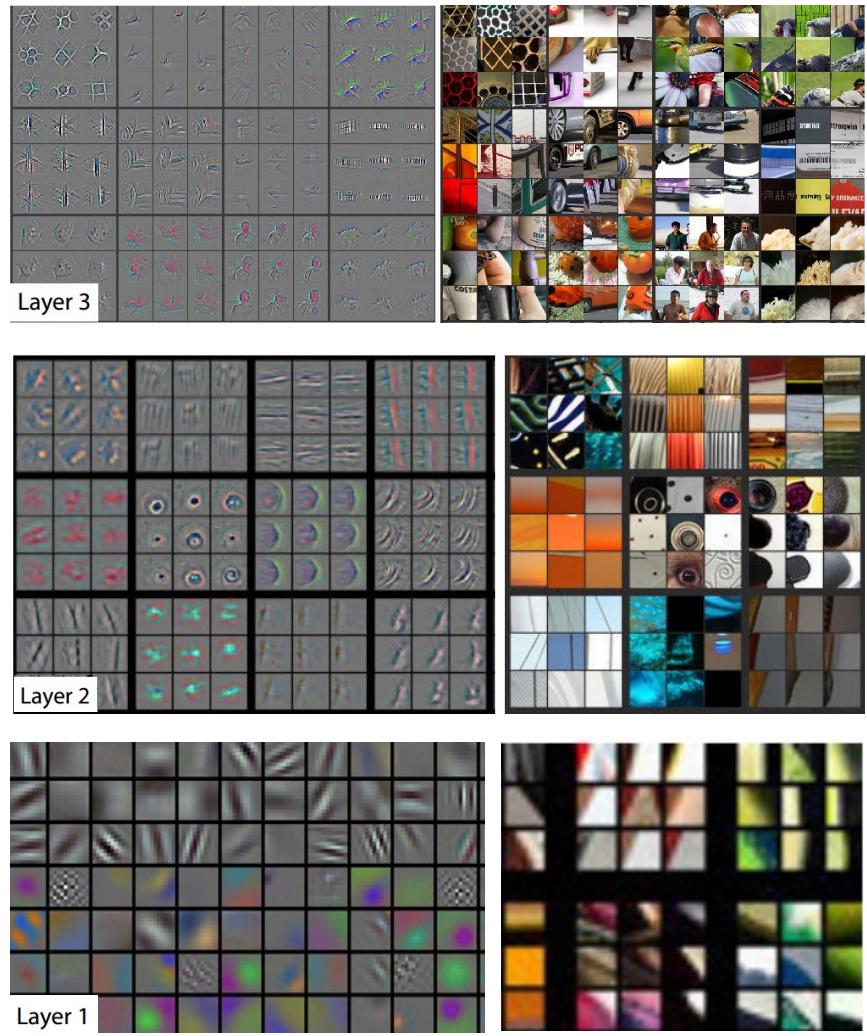
- The first breakthrough results of “deep learning” on large datasets happened in speech recognition
- Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition  
Dahl et al. (2010)

Acoustic model	Recog \\ WER	RT03S FSH	Hub5 SWB
Traditional features	1-pass –adapt	<b>27.4</b>	<b>23.6</b>
Deep Learning	1-pass –adapt	<b>18.5</b> (-33%)	<b>16.1</b> (-32%)



# Deep Learning for Computer Vision

- Most deep learning groups have (until 2 years ago) focused on computer vision
- Break through paper:  
ImageNet Classification with Deep Convolutional Neural Networks by Krizhevsky et al. 2012



Zeiler and Fergus (2013)

# Deep Learning + NLP = Deep NLP

- Combine ideas and goals of NLP and use representation learning and deep learning methods to solve them
- Several big improvements in recent years across different NLP
  - **levels:** speech, morphology, syntax, semantics
  - **applications:** machine translation, sentiment analysis and question answering

# Representations at NLP Levels: Phonology

- Traditional: Phonemes

CONSONANTS (PULMONIC)

© 2005 IPA

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		t d	c ɟ	k g	q ɢ		ʔ
Nasal	m	n̪		n		n̪	n̪	n̪	N		
Trill	B			r					R		
Tap or Flap		v̪		f		t̪					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	s z	ç j	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative			ɬ ɺ								
Approximant		v̪		x		ɻ	j	w̪			
Lateral approximant			l̪		ɭ	ɻ	L				

Where symbols appear in pairs, the one to the right represents a voiced consonant. Shaded areas denote articulations judged impossible.

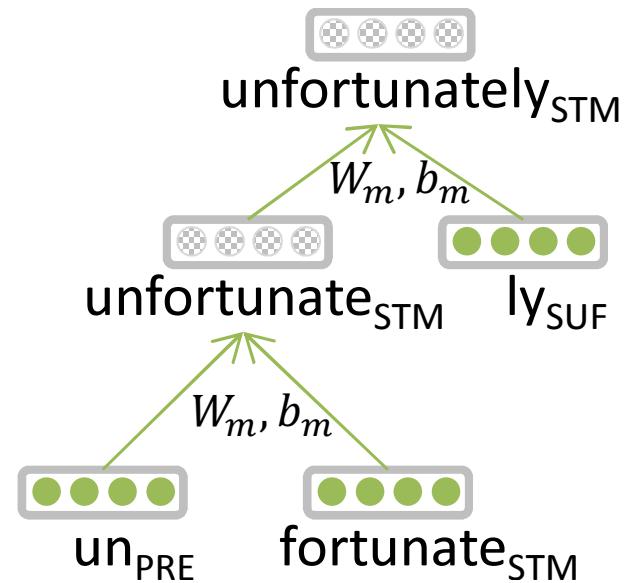
- DL: trains to predict phonemes (or words directly) from sound features and represent them as **vectors**

# Representations at NLP Levels: Morphology

- Traditional: Morphemes

prefix stem suffix  
un interest ed

- DL:
  - every morpheme is a vector
  - a neural network combines two vectors into one vector
  - Thang et al. 2013

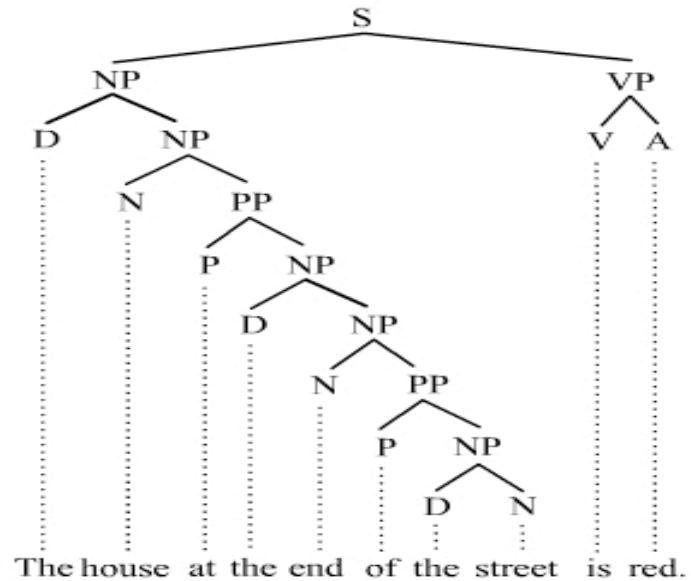


# Neural word vectors - visualization

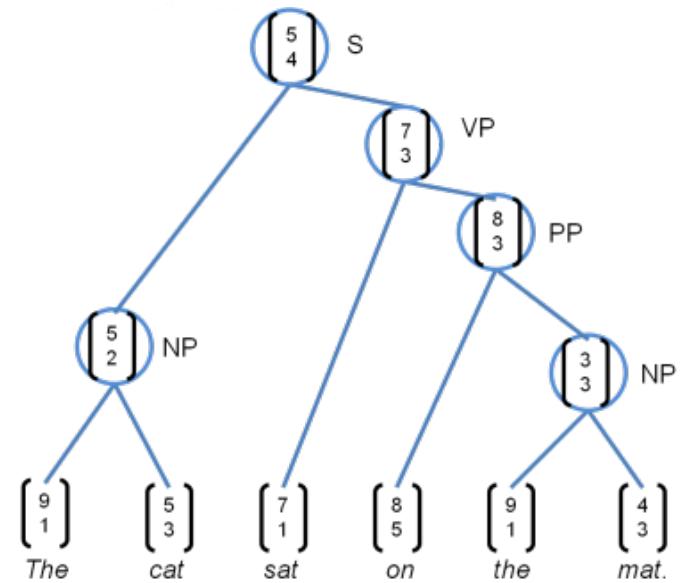


# Representations at NLP Levels: Syntax

- Traditional: Phrases  
Discrete categories like NP, VP

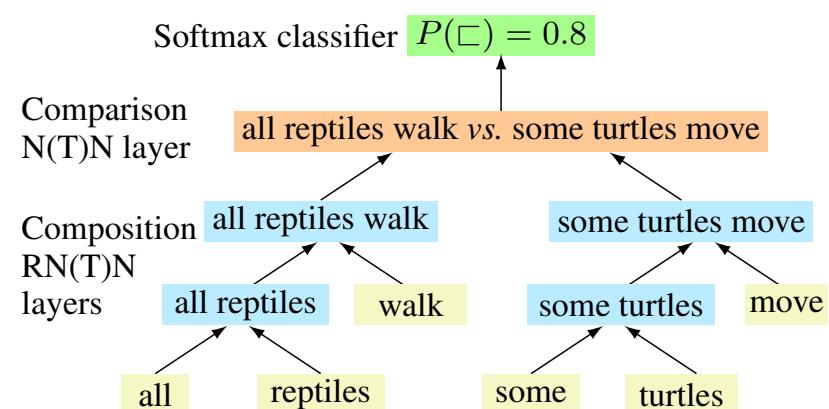
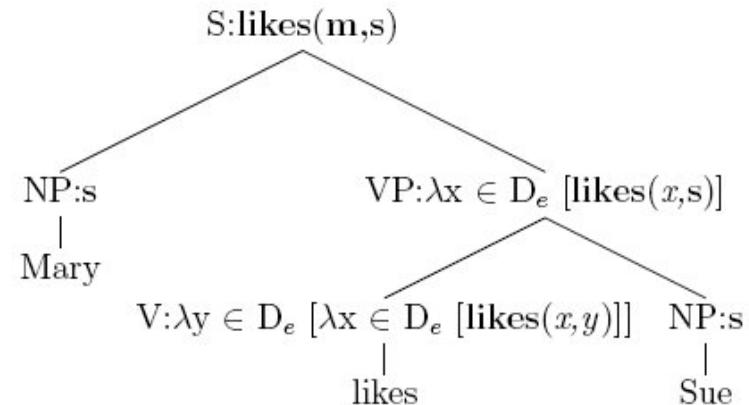


- DL:
  - Every word and every phrase is a vector
  - a neural network combines two vectors into one vector
  - Socher et al. 2011



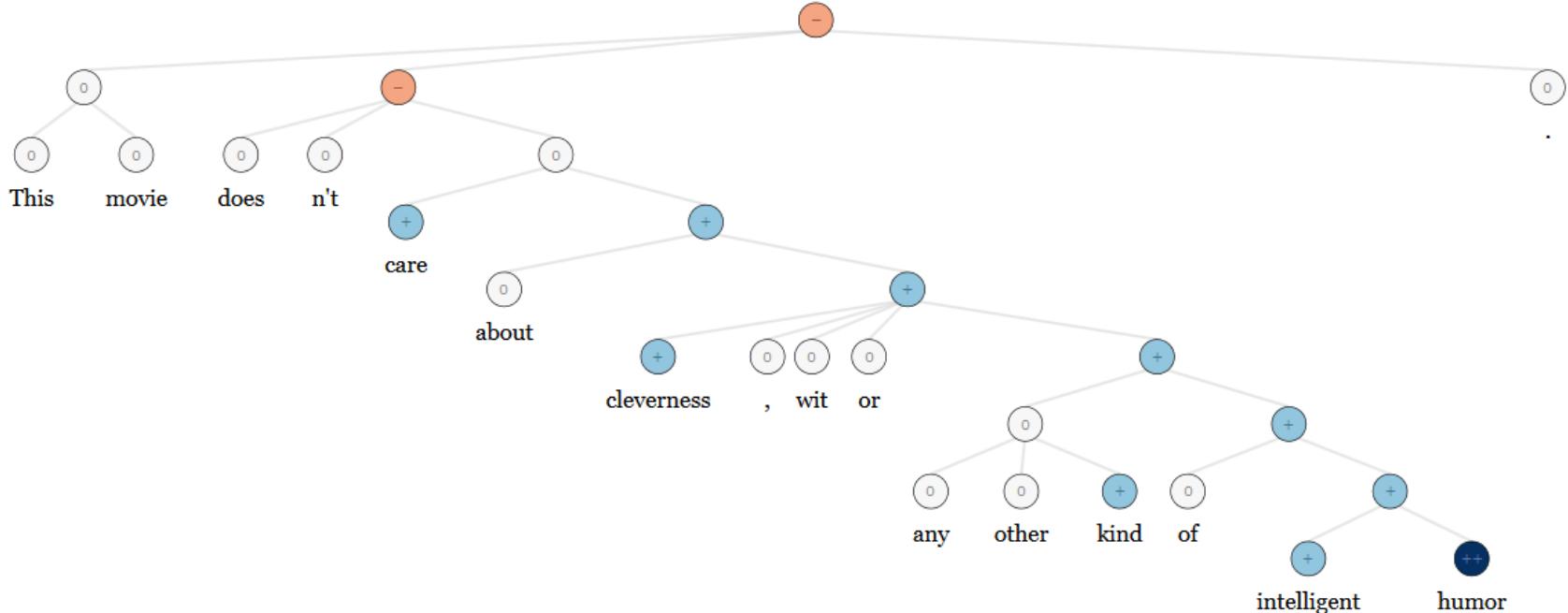
# Representations at NLP Levels: Semantics

- Traditional: Lambda calculus
  - Carefully engineered functions
  - Take as inputs specific other functions
  - No notion of similarity or fuzziness of language
- DL:
  - Every word and every phrase and every logical expression is a vector
  - a neural network combines two vectors into one vector
  - Bowman et al. 2014



# NLP Applications: Sentiment Analysis

- Traditional: Curated sentiment dictionaries combined with either bag-of-words representations (ignoring word order) or hand-designed negation features (ain't gonna capture everything)
- Same deep learning model that was used for morphology, syntax and logical semantics can be used! → RecursiveNN

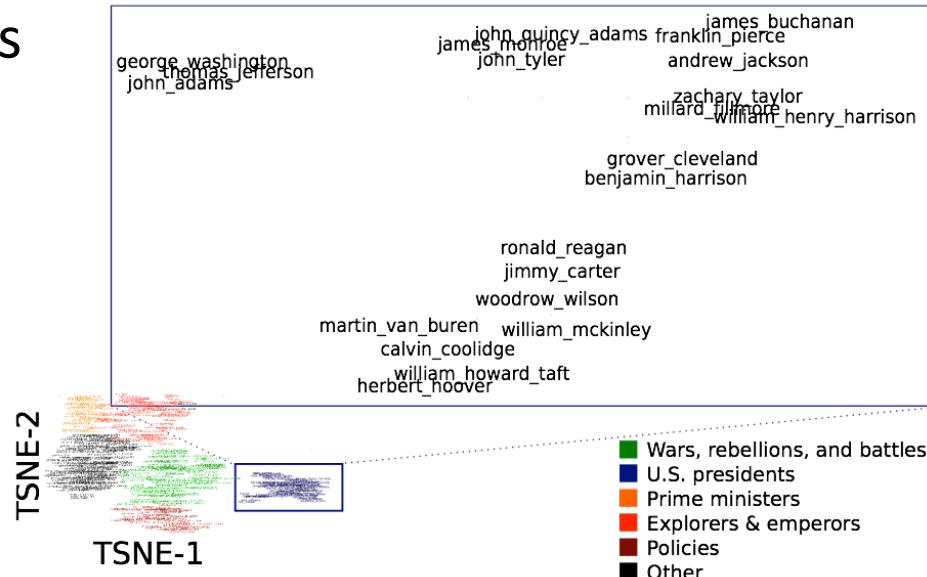


# Question Answering

- Common: A lot of feature engineering to capture world and other knowledge, e.g. regular expressions, Berant et al. (2014)

Is main verb trigger?	
Yes	No
Condition	Regular Exp.
Wh- word subjective?	AGENT
Wh- word object?	THEME
default	(ENABLE SUPER) <sup>+</sup>
DIRECT	(ENABLE SUPER)
PREVENT	(ENABLE SUPER)* PREVENT(ENABLE SUPER)*

- DL: Same deep learning model that was used for morphology, syntax, logical semantics and sentiment can be used!
- Facts are stored in vectors



# Machine Translation

- Many levels of translation have been tried in the past:
- Traditional MT systems are very large complex systems

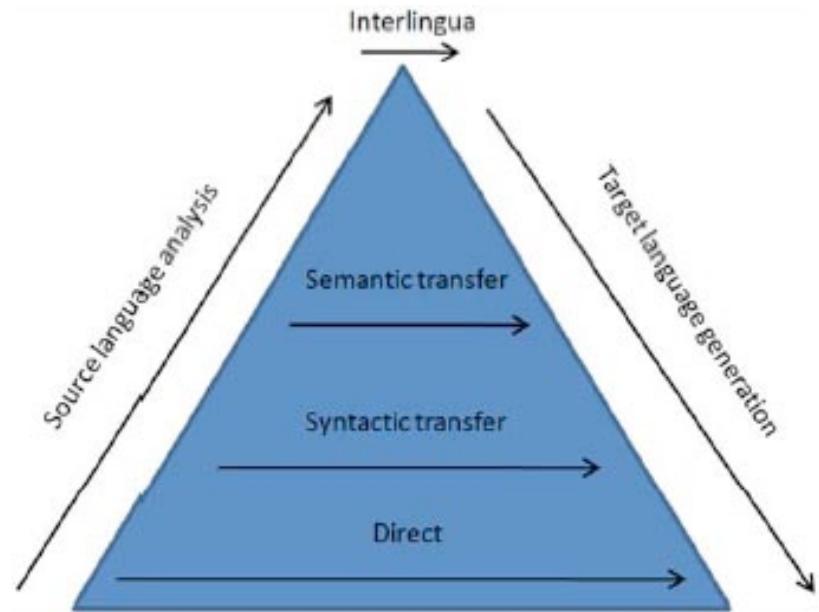


Figure 1: The Vauquois triangle

- What do you think is the interlingua for the DL approach to translation?

# Machine Translation

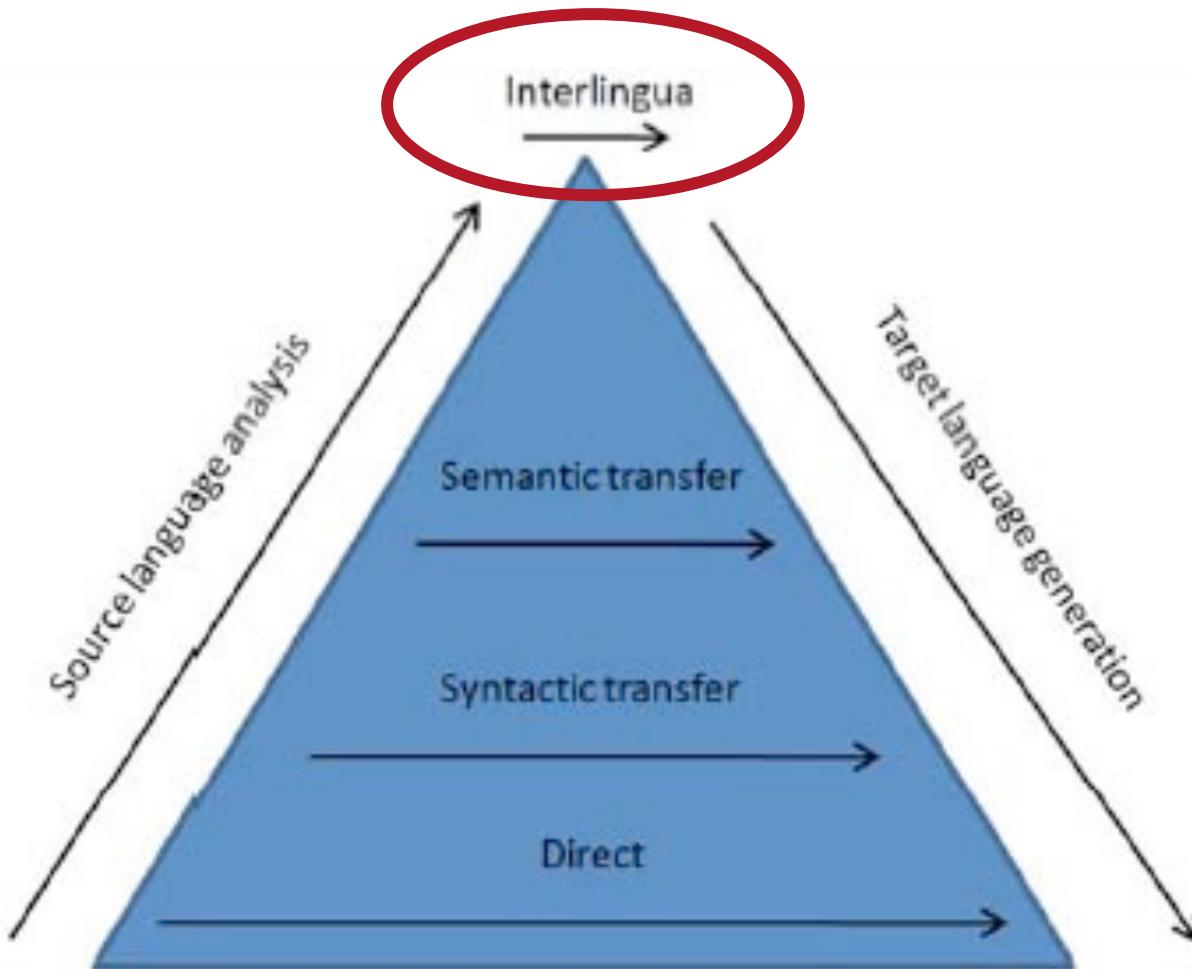
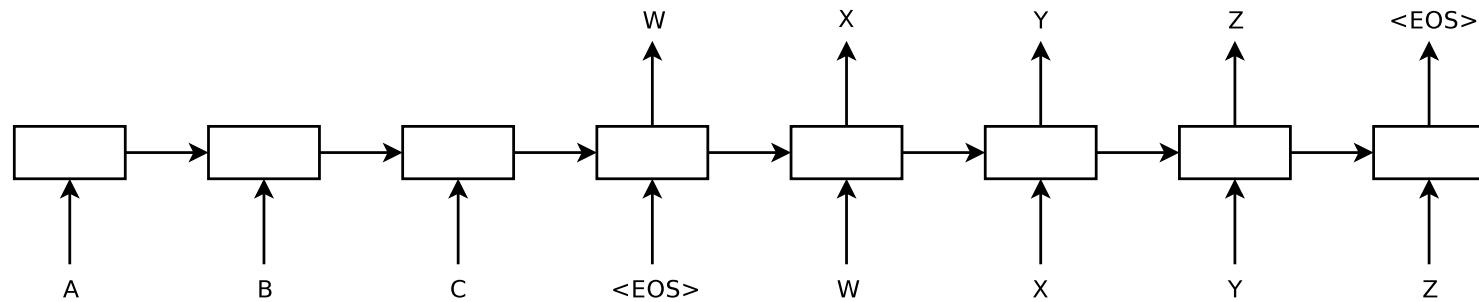


Figure 1: The Vauquois triangle

# Machine Translation

- Source sentence mapped to vector, then output sentence generated.



- Sequence to Sequence Learning with Neural Networks by Sutskever et al. 2014; Luong et al. 2016
- About to replace very complex hand engineered architectures

# Dynamic Memory Network by MetaMind

## Story

The best way to hope for any chance of enjoying this film is by lowering your expectations.

## Question

What is the sentiment?

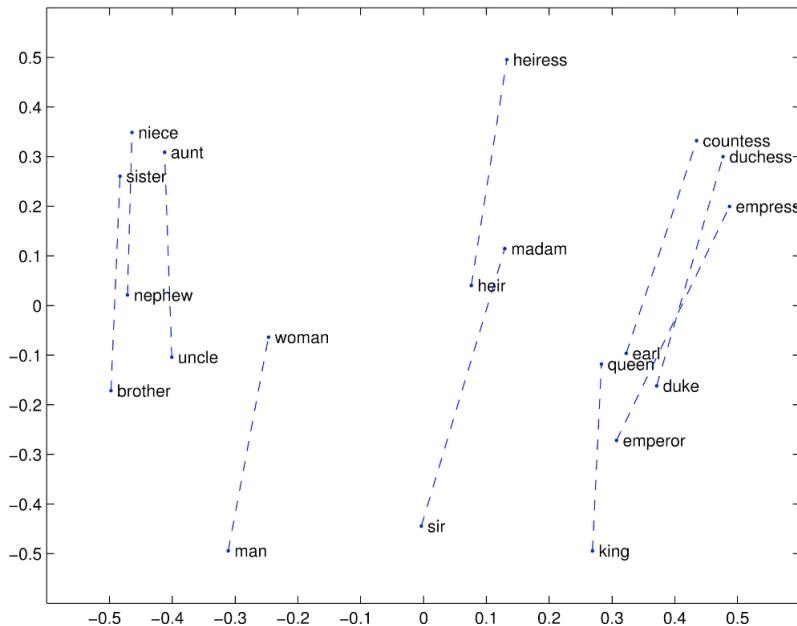
Run DMN



Get new example

# Representation for all levels: Vectors

- We will learn in the next lecture how we can learn vector representations for words and what they actually represent.



- Next week: neural networks and how they can use these vectors for all NLP levels and many different applications