Deep Learning and Continuous Representations for Natural Language Processing

Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao
Microsoft Research

IJCAI July 9th, 2016

Tutorial Outline

Jianfeng Gao

- Part I: Background
- Part II: Deep learning in statistical machine translation and conversation

Xiaodong He

- Part III: Continuous representations for selected NLP tasks

Scott Yih

- Part IV: Natural language understanding
- Part V: Conclusion

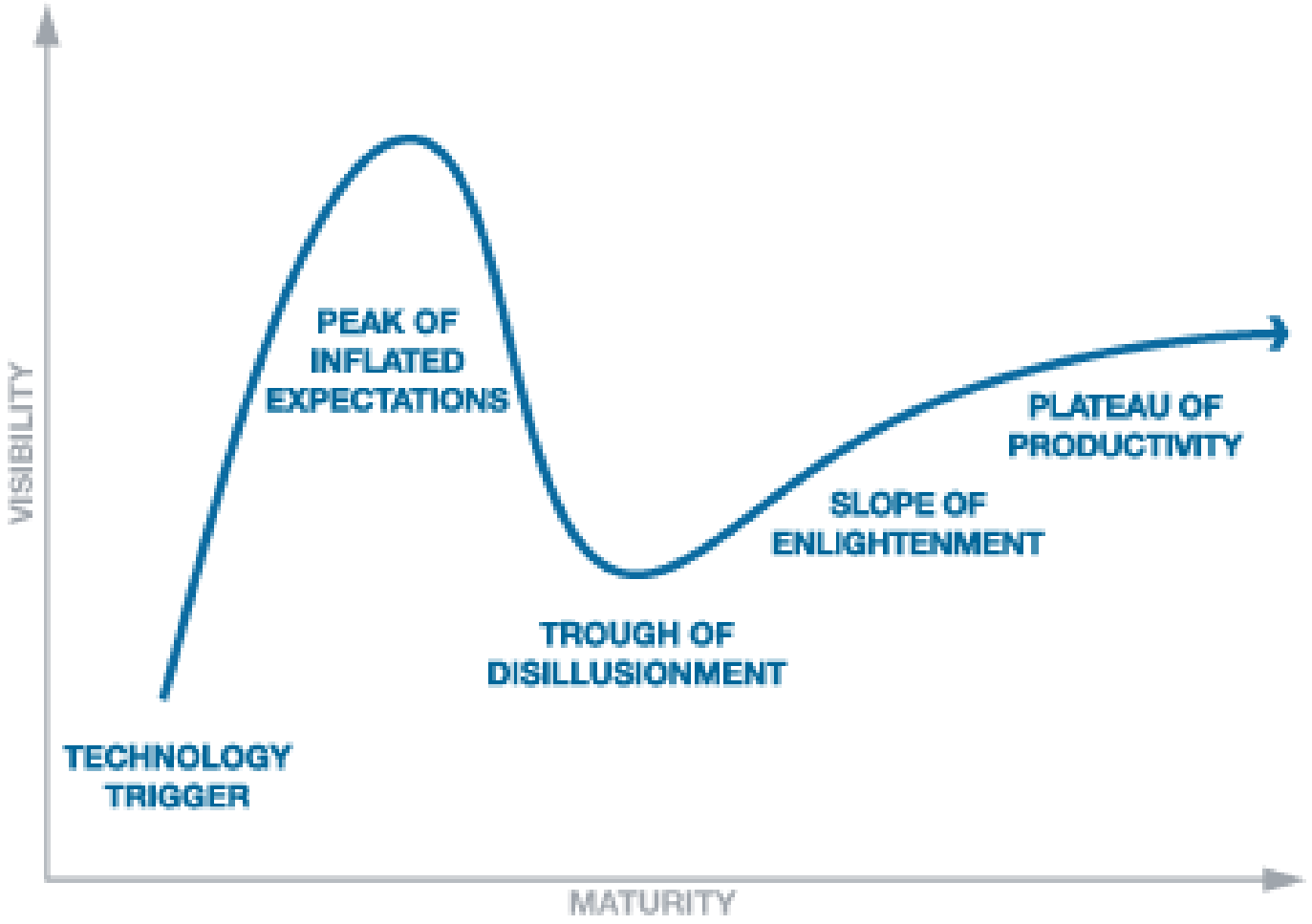
Part I

Background

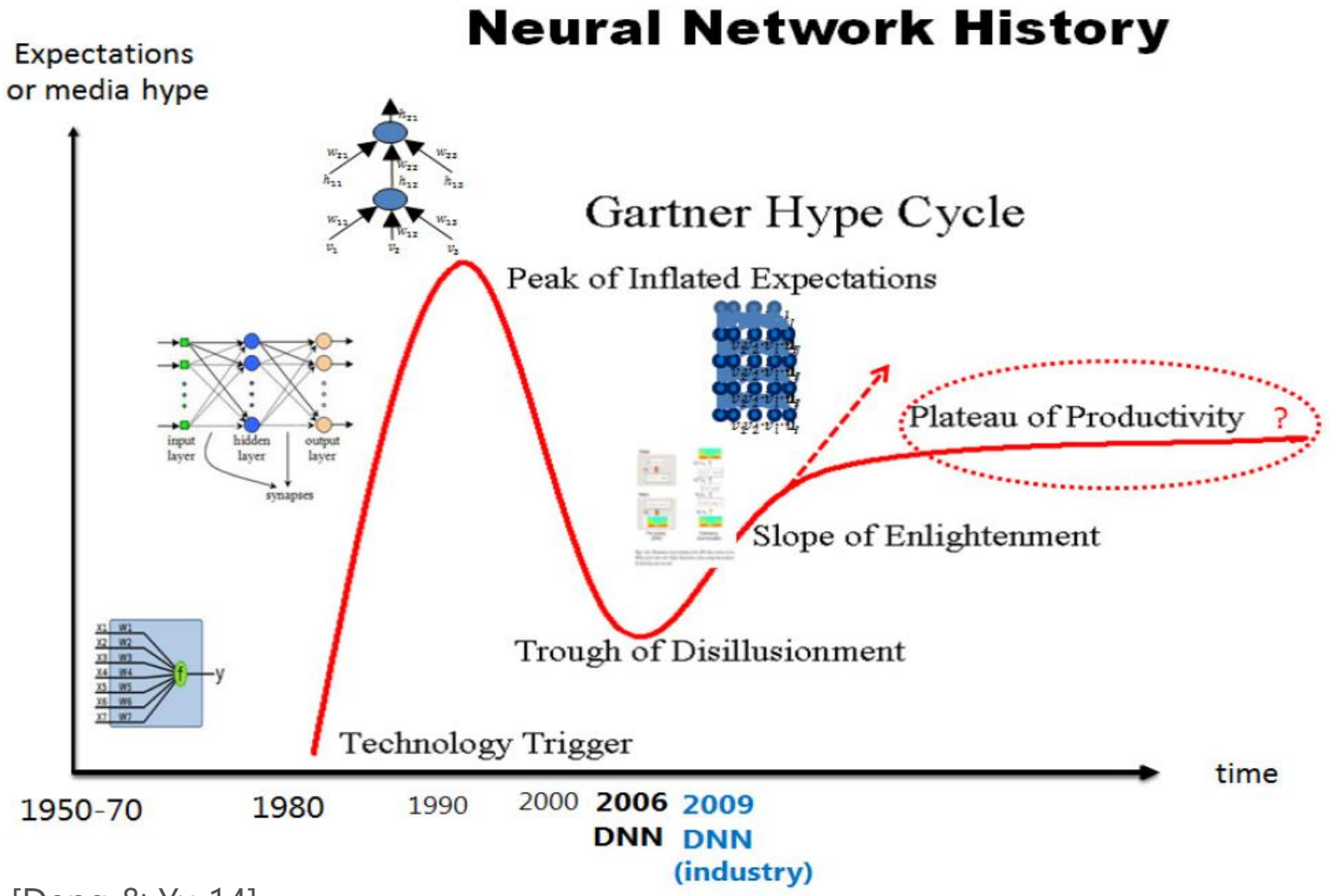
Tutorial Outline

- Part I: Background
  - A brief history of deep neural networks (DNN)
  - An example of neural models for query classification
  - Different forms of DNN for classification/ranking/generation tasks
- Part II: Deep learning in statistical machine translation and conversation
- Part III: Continuous representations for selected NLP tasks
- Part IV: Natural language understanding
- Part V: Conclusion

Gartner hype cycle



A brief history of deep neural networks (DNN)



[Deng & Yu 14]

MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Introduction The 10 Technologies Past Years

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Temporary Social Media

Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.

Prenatal DNA Sequencing

Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?

Additive Manufacturing

Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.

Baxter: The Blue-Collar Robot

Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.

Memory Implants

A maverick neuroscientist believes he has deciphered the code by which the brain forms long-term memories. Next: testing a prosthetic implant for people suffering from long-term memory loss.

Smart Watches

The designers of the Pebble watch realized that a mobile phone is more useful if you don't have to take it out of your pocket.

Ultra-Efficient Solar Power

Doubling the efficiency of a solar cell would completely change the economics of renewable energy. Nanotechnology just might make it possible.

Big Data from Cheap Phones

Collecting and analyzing information from simple cell phones can provide surprising insights into how people move about and behave – and even help us understand the spread of diseases.

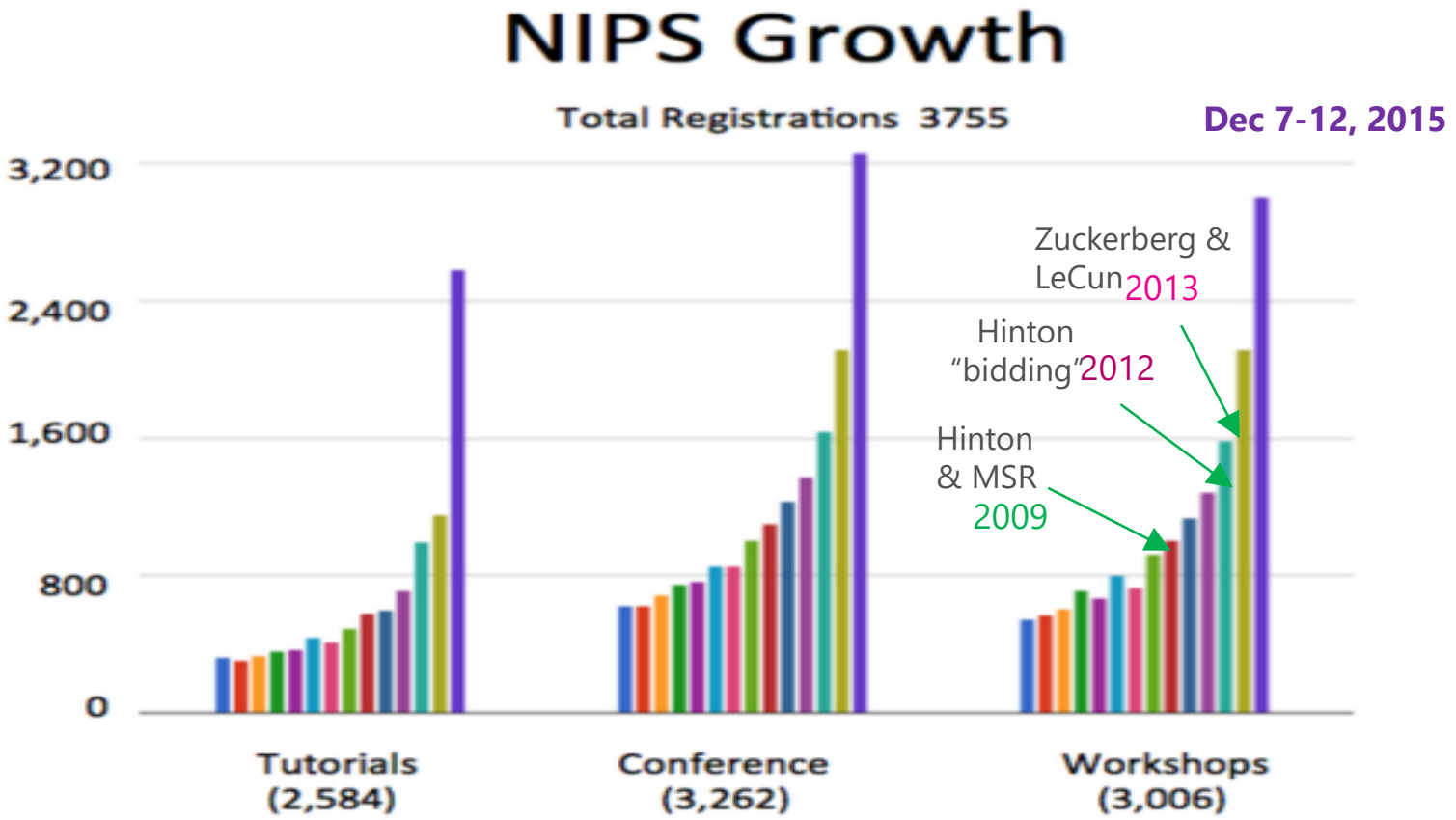
Supergrids

A new high-power circuit breaker could finally make highly efficient DC power grids practical.

Deep learning in academia: centered at NIPS 2015

Neil Lawrence @lawrennd · Dec 7

#NIPS2015 attendance numbers. Massive growth across the board but over 100% in tutorials.



Geoff Hinton

The universal translator on “Star Trek” comes true…

The New York Times

Scientists See Promise in Deep-Learning Programs

John Markoff November 23, 2012

Rick Rashid in **Tianjin, China**, October, 25, 2012

A voice recognition program translated a speech given by Richard F. Rashid, Microsoft’s top scientist, into Chinese.

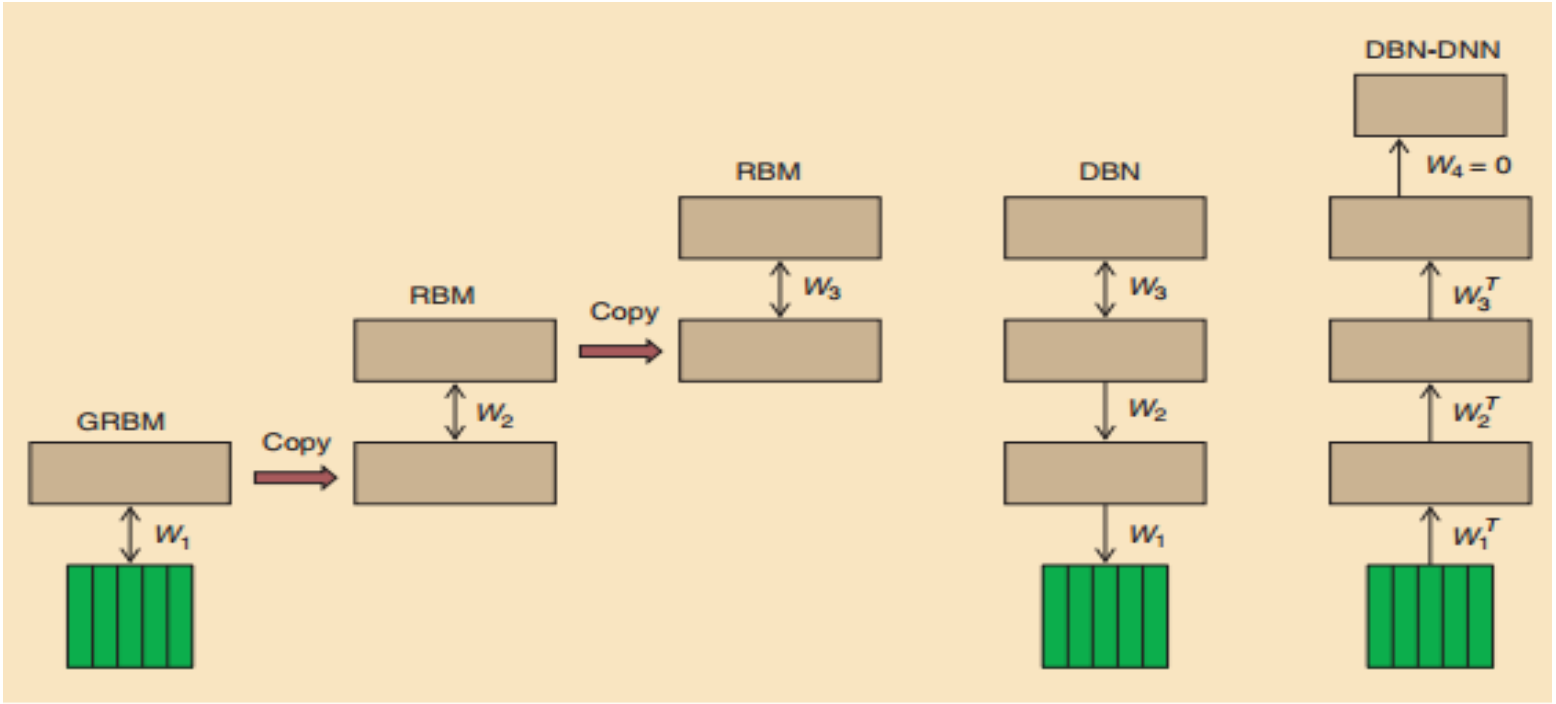
Geoff Hinton

Li Deng

Dong Yu

DNN: (Fully-Connected) Deep Neural Networks

Hinton, Deng, Yu, et al., DNN for AM in speech recognition, *IEEE SPM*, 2012

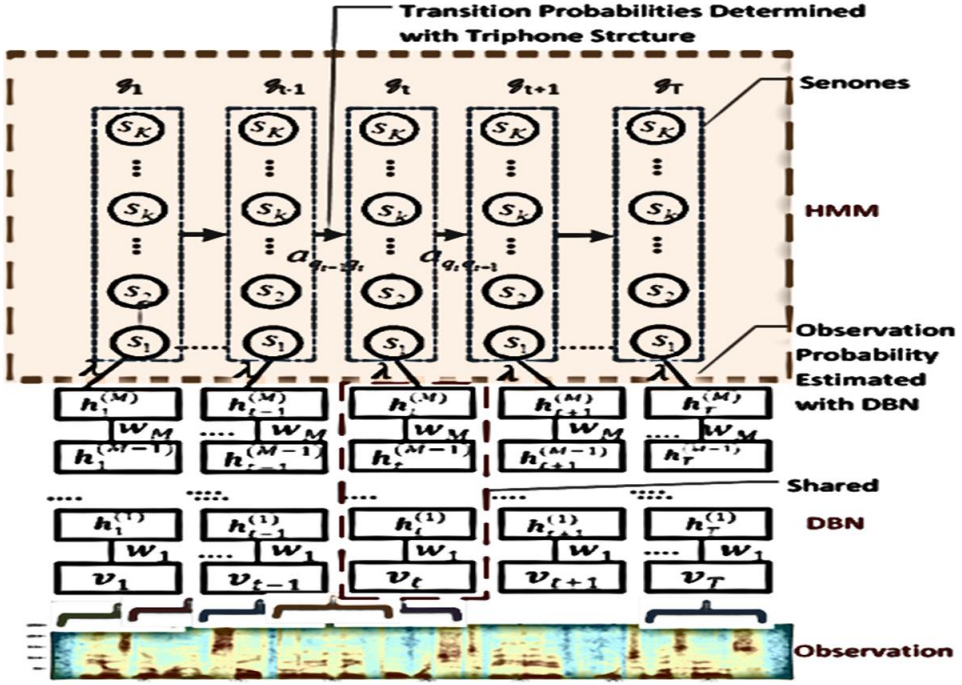


First train a stack of N models each of which has one hidden layer. Each model in the stack treats the hidden variables of the previous model as data.

Then compose them into a single Deep Belief Network.

Then add outputs and train the DNN with backprop.

CD-DNN-HMM

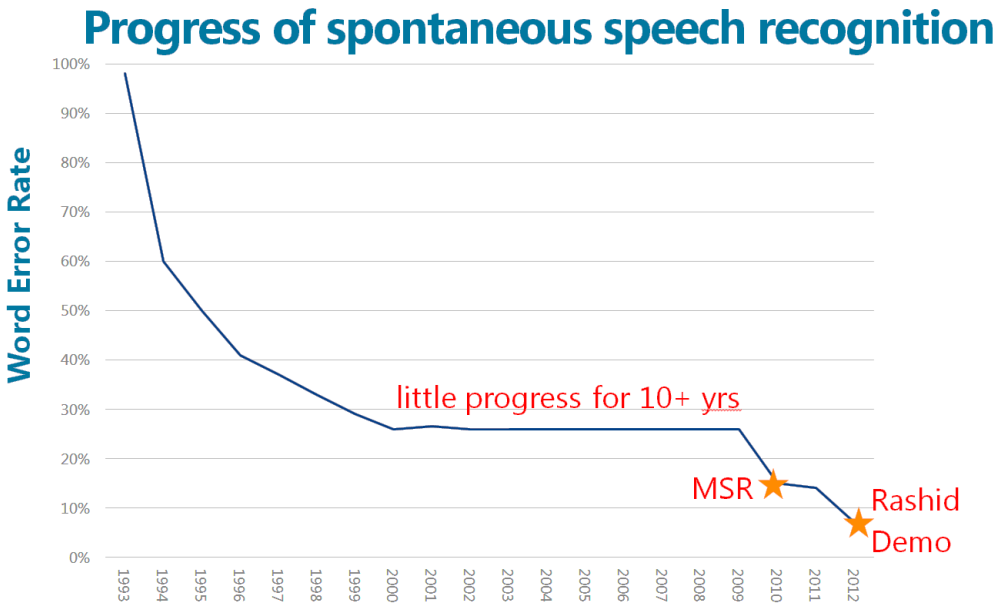


Dahl, Yu, Deng, and Acero, "Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Trans. ASLP*, Jan. 2012

Seide, Li, and Yu, "Conversational Speech Transcription using Context-Dependent Deep Neural Networks," *INTERSPEECH* 2011.

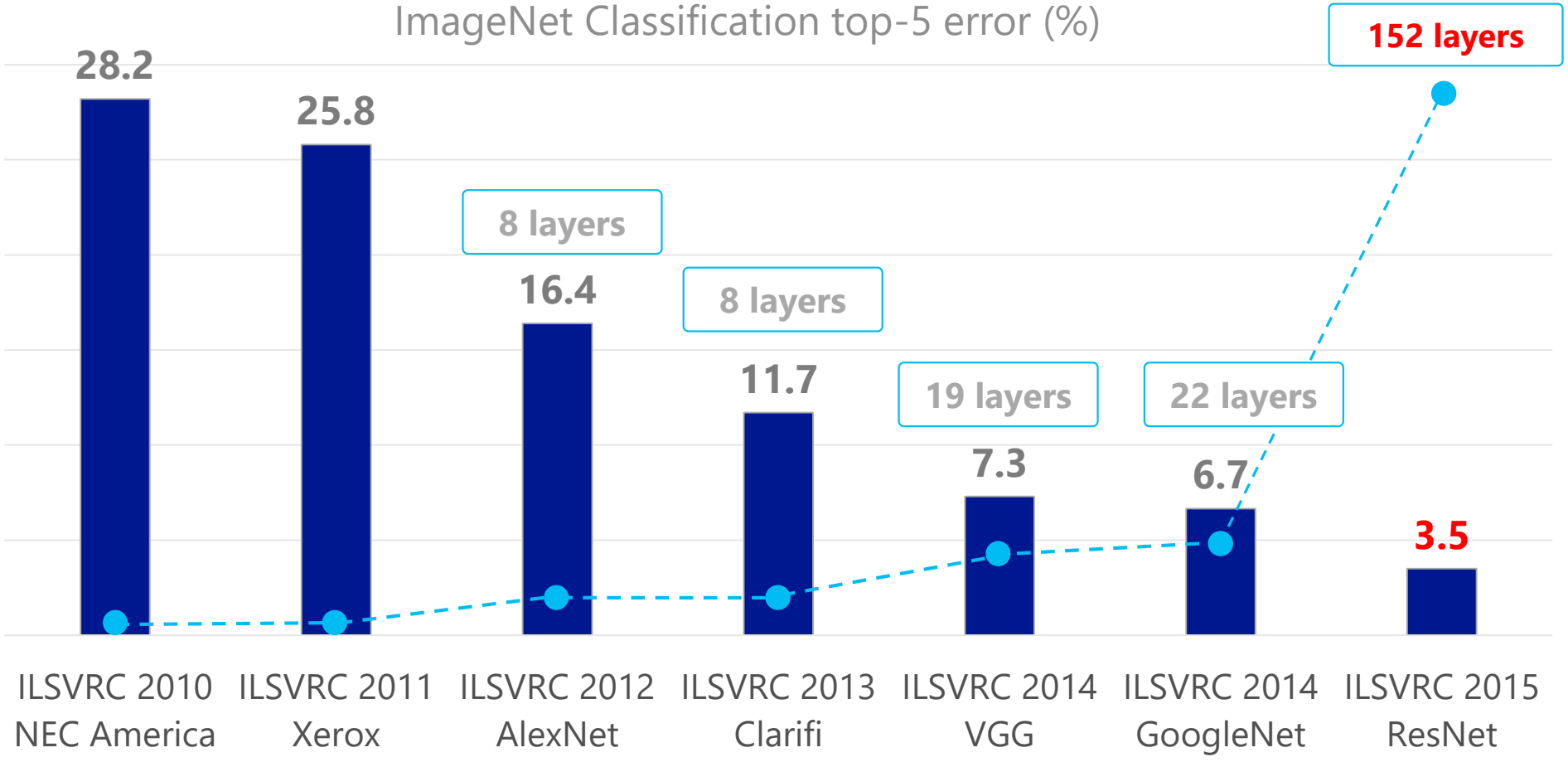
After no improvement for 10+ years by the research community…

MSR reduced error from **~23%** to **<13%** (and under 7% for Rick Rashid's S2S demo)!



ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

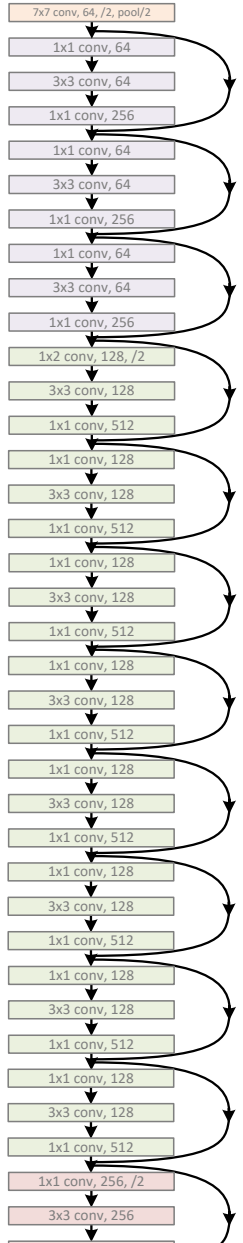
ImageNet Large Scale Visual Recognition Challenge (ILSVRC)



Microsoft

Slide from Jian Sun

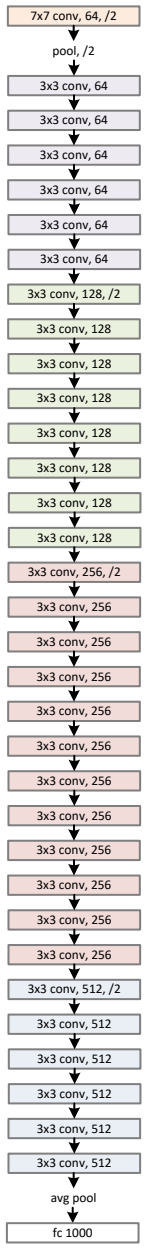
Revolution of Depth: ResNet w. 152 layers



Slide from Jian Sun

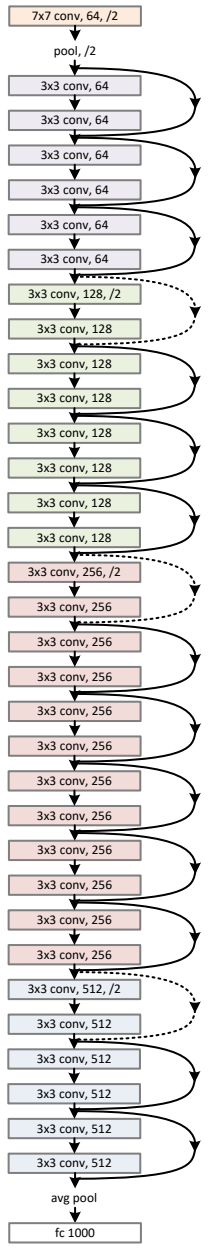
Plain deep network

signals flow thru a single path



Deep residual network

signals flow thru many paths



Kaiming He, Xiangyu Zhang, Shaoqing Ren, & Jian Sun. "Deep Residual Learning for Image Recognition". arXiv 2015.

The focus of this tutorial

- Is not on speech or image,
- But on text processing and understanding tasks
  - Statistical machine translation
  - Conversation
  - Information retrieval
  - Image captioning
  - Question answering
  - Etc.

A query classification problem

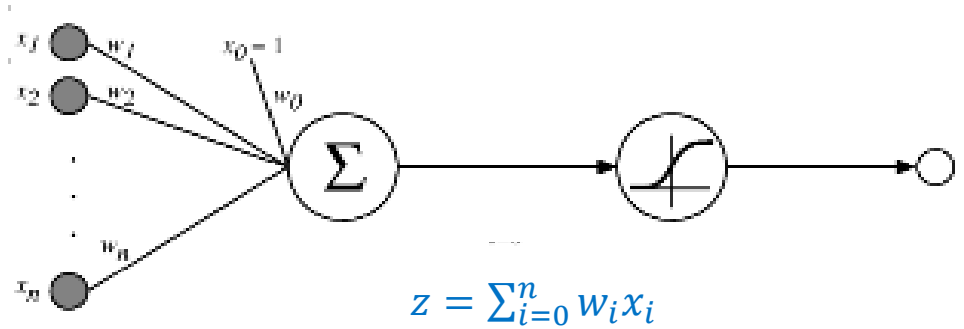
- Given a search query q, e.g., "denver sushi downtown"
- Identify its domain c e.g.,
  - Restaurant
  - Hotel
  - Nightlife
  - Flight
  - etc.
- So that a search engine can tailor the interface and result to provide a richer personalized user experience

A single neuron model

- For each domain c, build a binary classifier
  - Input: represent a query q as a vector of features $x = [x_1, \dots x_n]^T$
  - Output: $y = P(1|q, c)$
  - q is labeled c if $P(1|q, c) > 0.5$
- Input feature vector, e.g., a bag of words vector
  - Regards words as atomic symbols: *denver, sushi, downtown*
  - Each word is represented as a one-hot vector: $[0, \dots, 0, 1, 0, \dots, 0]^T$
  - Bag of words vector = sum of one-hot vectors
  - We may use other features, such as n-grams, phrases, (hidden) topics

A single neuron model

Input features x

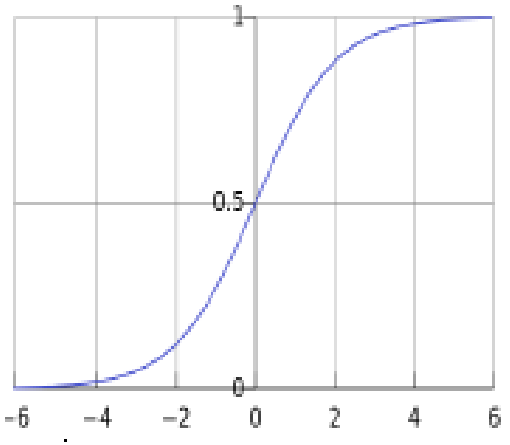


$z = \sum_{i=0}^{n} w_i x_i$

Output: $P(1|q, c)$

$y = \sigma(z) = \frac{1}{1+\exp(-z)}$

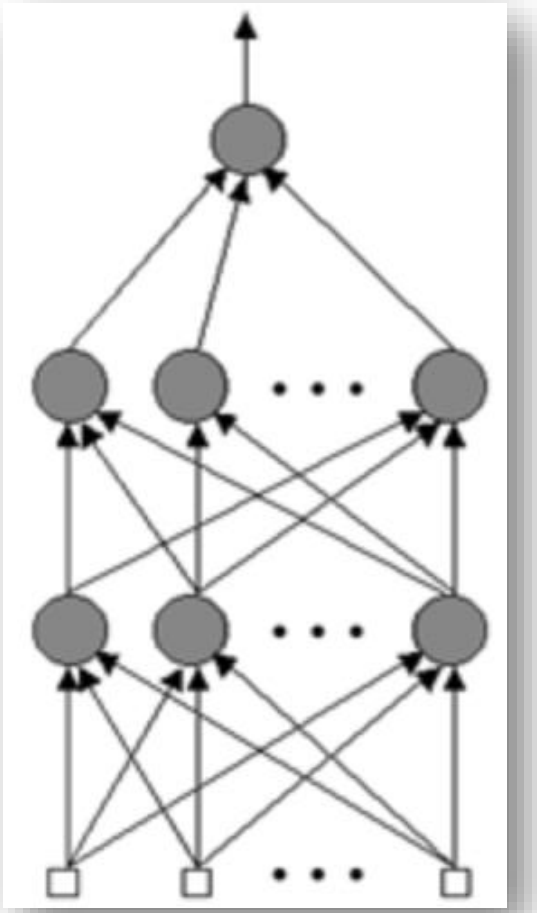
- w: weight vector to be learned
- z: weighted sum of input features
- σ: the logistic function
  - Turn a score to a probability
  - non-linear activation function, essential in DNN models



Model training: how to assign $\boldsymbol{w}$

- Training data: a set of $\left(x^{(m)}, y^{(m)}\right)_{m=\{1,2,\ldots,M\}}$ pairs
  - Input $x^{(m)} \in R^n$
  - Output $y^{(m)} = \{0,1\}$
- optimize parameters $\boldsymbol{w}$ on training data
  - minimize a loss function (mean square error loss)
    - $\min_w \sum_{m=1}^{M} L^m$
    - where $L^{(m)} = \frac{1}{2}\left(f_w\left(x^{(m)}\right) - y^{(m)}\right)^2$
  - Using Stochastic Gradient Descent (SGD)
    - Initialize w randomly
    - Update for each training sample until convergence: $w^{new} = w^{old} - \eta \frac{\partial L}{\partial w}$

Multi-layer (deep) neural networks



Output layer $y^o = \sigma(w^T y^2)$

Vector w

2^{st} hidden layer $y^2 = \sigma(\mathbf{W}_2 y^1)$

This is exactly the **single neuron model** with **hidden** features.

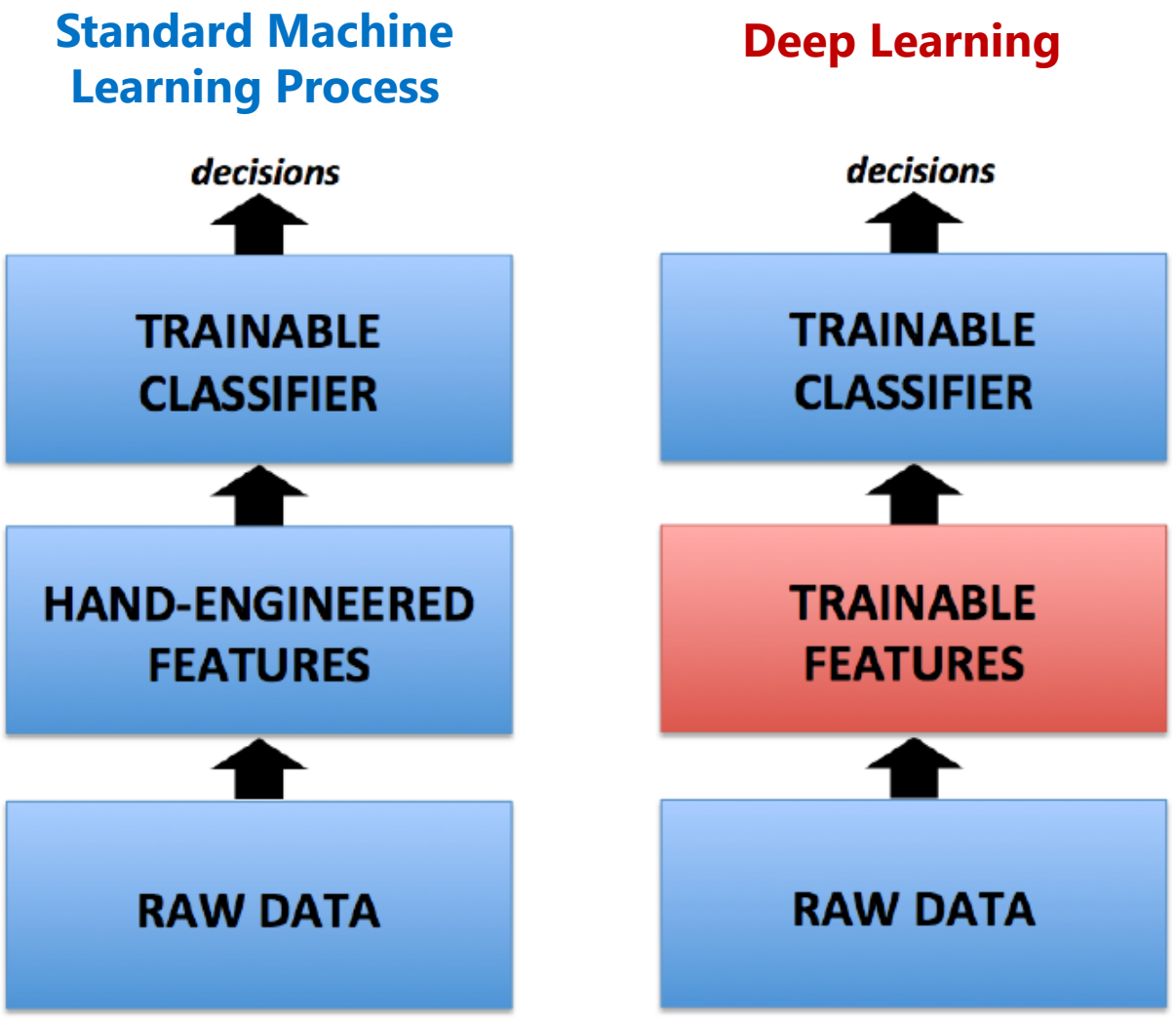
Projection matrix $\mathbf{W}_2$

1^{st} hidden layer $y^1 = \sigma(\mathbf{W}_1 x)$

Feature generation: project raw input features (bag of words) to **hidden** features (topics).

Projection matrix $\mathbf{W}_1$

Input features x

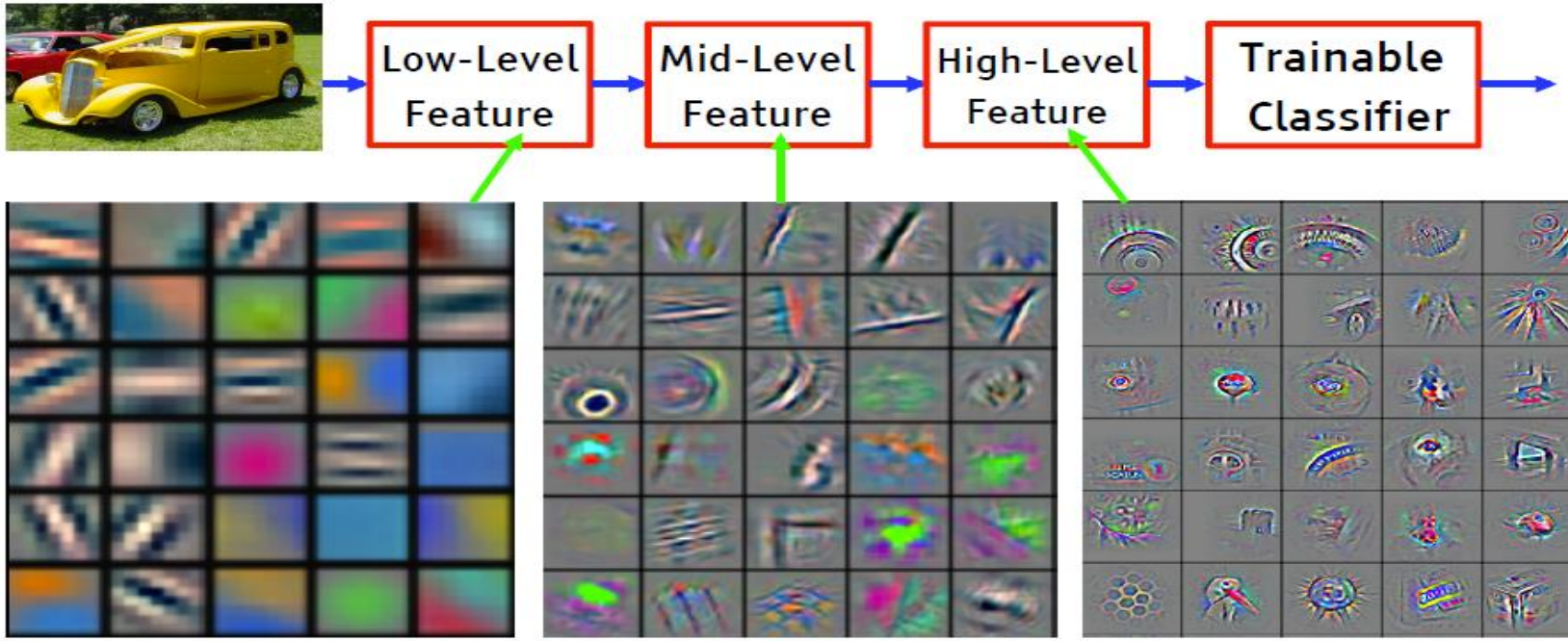


Adapted from [Duh 14]

Why Multiple Layers?

DL tutorial at NIPS'2015

- Hierarchy of representations with increasing level of abstraction
- Each layer is a trainable feature transform
- Image recognition: pixel → edge → texton → motif → part → object
- ?? Text: character → word → word group → clause → sentence → story



Different forms of DNN

- Classification task – label X by Y
  - Multi-Layer Perceptron
  - Convolutional NN
- Ranking task – compute the sim btw X and Y
  - Siamese neural network [Bromley et al. 1993]
  - Deep Semantic Similarity Model (DSSM)
- (Text) Generation task – generate Y from X
  - Seq2Seq (RNN/LSTM)
  - Memory Network

Deep Semantic Similarity Model (DSSM)

[Huang+ 13; Gao+ 14a; Gao+ 14b; Shen+ 14; Yih+ 15; Fang+15]

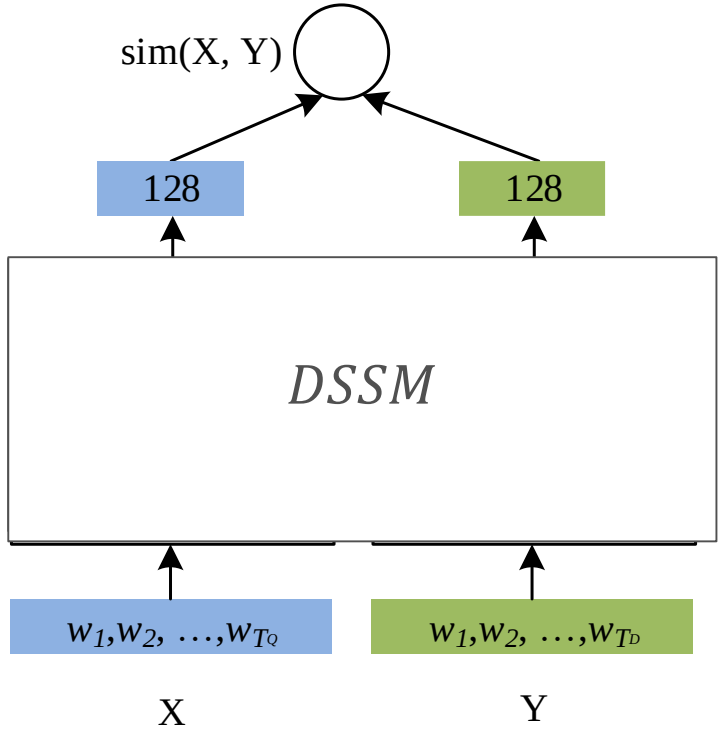
- Compute semantic similarity btw text strings X and Y
  - Map X and Y to feature vectors in a latent semantic space via deep neural net
  - Compute the cosine similarity between the feature vectors
  - Also called "Deep Structured Similarity Model" in [Huang+ 13]

Tasks	X	Y	Ref
Machine translation	*Text in language A*	*Translation in language B*	[Gao+ 14a]
Web search	*Search query*	*Web document*	[Huang+ 13; Shen+ 14]
Image captioning	*Image*	*Text caption*	[Fang+ 15]
Question Answering	*Question*	*Answer*	[Yih+ 15]
Contextual entity linking	*Mention (in text)*	*Entities (in Satori)*	[Gao+ 14b]
Ad selection	*Search query*	*Ad keywords*	
…	…	…	

Sent2Vec (DSSM) http://aka.ms/sent2vec

DSSM: Compute Similarity in Semantic Space

Relevance measured by cosine similarity

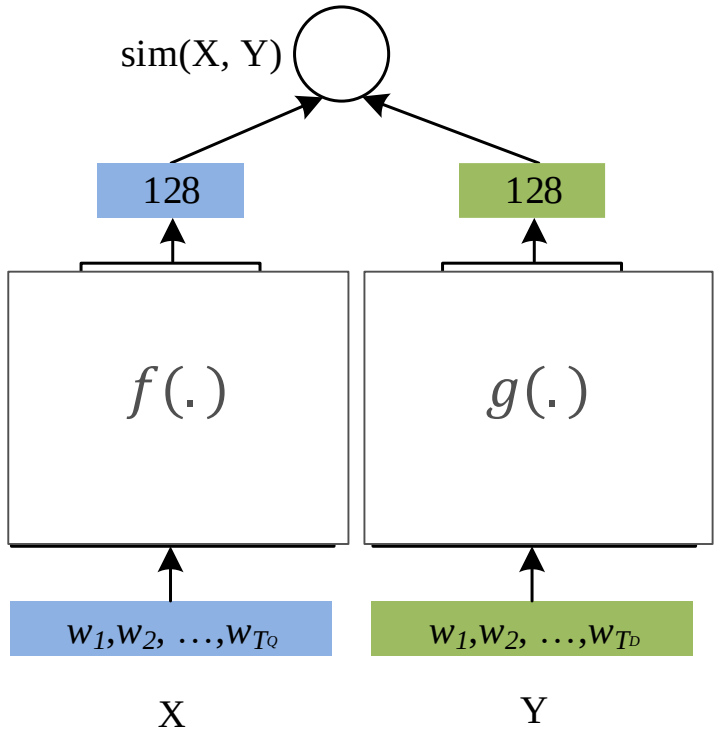


Word sequence x_t

Learning: maximize the similarity between X (source) and Y (target)

DSSM: Compute Similarity in Semantic Space

Relevance measured by cosine similarity



Word sequence x_t

Learning: maximize the similarity between X (source) and Y (target)

Representation: use DNN to extract abstract semantic representations

Convolutional DSSM [Gao+ 14b; Shen+ 14]

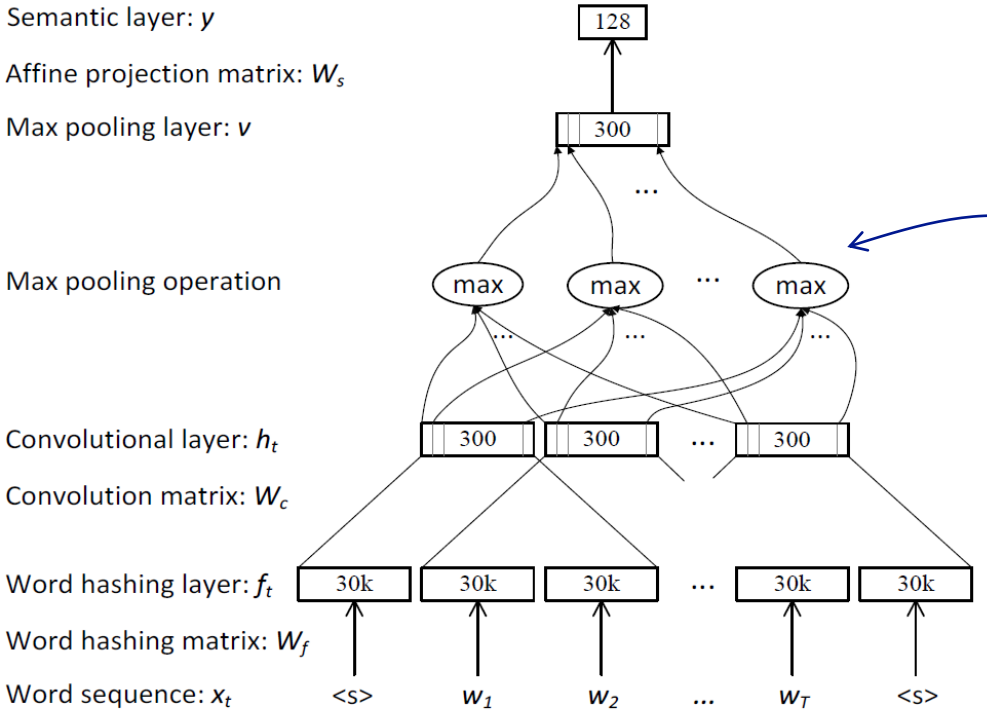
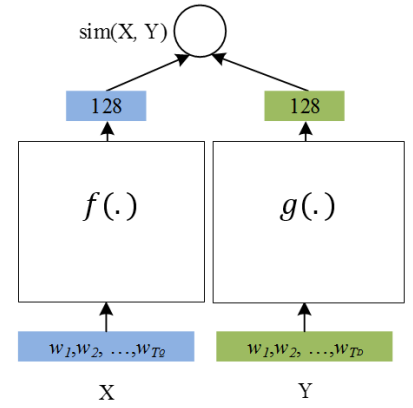


Figure 1: Illustration of the C-DSSM. A convolutional layer with the window size of three is illustrated.

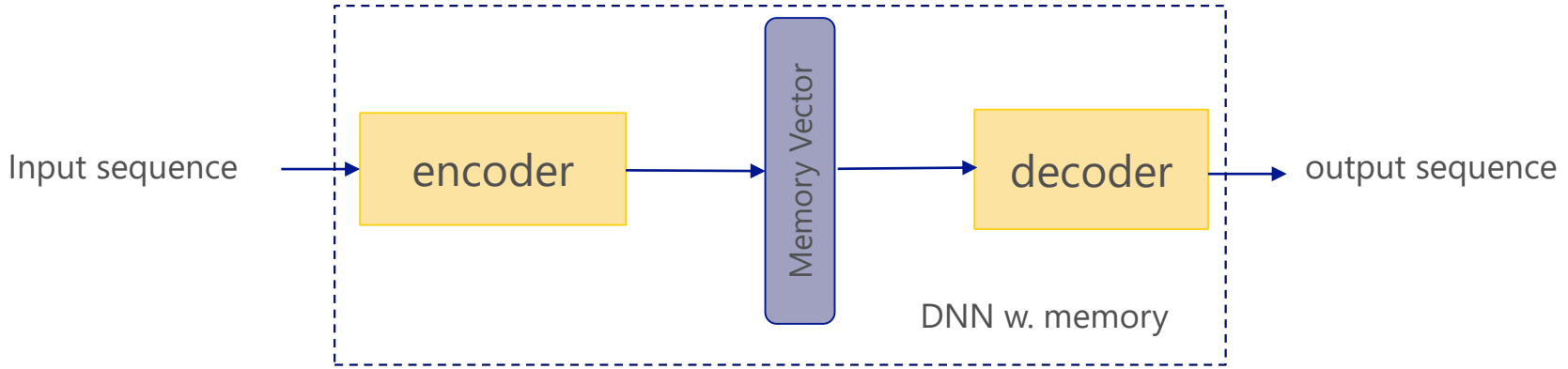
Model local context at the convolutional layer

Model global context at the pooling layer

Identify key words/concepts in X (and Y)



Sequence-to-Sequence Tasks [Sutskever+ 14]



- Statistical Machine translation (SMT):
  - A sentence in source language → A sentence in target language
- Conversation (chitchat):
  - Context + message → response
- *Question answering + recommendation dialog:*
  - *Knowledge base + context + question → answer/recommendation*

QA + Recommendation Dialog [Dodge+ 16]

Information/sentences retrieved from Knowledge base.

Conversation context

Query

Long-Term Memories	h_i	Shaolin Soccer directed_by Stephen Chow Shaolin Soccer written_by Stephen Chow Shaolin Soccer starred_actors Stephen Chow Shaolin Soccer release_year 2001 Shaolin Soccer has_genre comedy Shaolin Soccer has_tags martial arts, kung fu soccer, stephen chow Kung Fu Hustle directed_by Stephen Chow Kung Fu Hustle written_by Stephen Chow Kung Fu Hustle starred_actors Stephen Chow Kung Fu Hustle has_genre comedy action Kung Fu Hustle has_imdb_votes famous Kung Fu Hustle has_tags comedy, action, martial arts, kung fu, china, soccer, hong kong, stephen chow The God of Cookery directed_by Stephen Chow The God of Cookery written_by Stephen Chow The God of Cookery starred_actors Stephen Chow The God of Cookery has_tags hong kong Stephen Chow From Beijing with Love directed_by Stephen Chow From Beijing with Love written_by Stephen Chow From Beijing with Love starred_actors Stephen Chow, Anita Yuen . . . <and more> . . .
Short-Term Memories	c_1^u c_1^r	1) I'm looking a fun comedy to watch tonight, any ideas? 2) Have you seen Shaolin Soccer? That was zany and great.. really funny but in a whacky way.
Input	c_2^u	3) Yes! Shaolin Soccer and Kung Fu Hustle are so good I really need to find some more Stephen Chow films I feel like there is more awesomeness out there that I haven't discovered yet ...
Output	y	4) God of Cookery is pretty great, one of his mid 90's hong kong martial art comedies.

End-to-End Memory Networks (MemNN)

[Sukhbaatar+ 15]

- Retrieving long-term mem x
- Embedding input

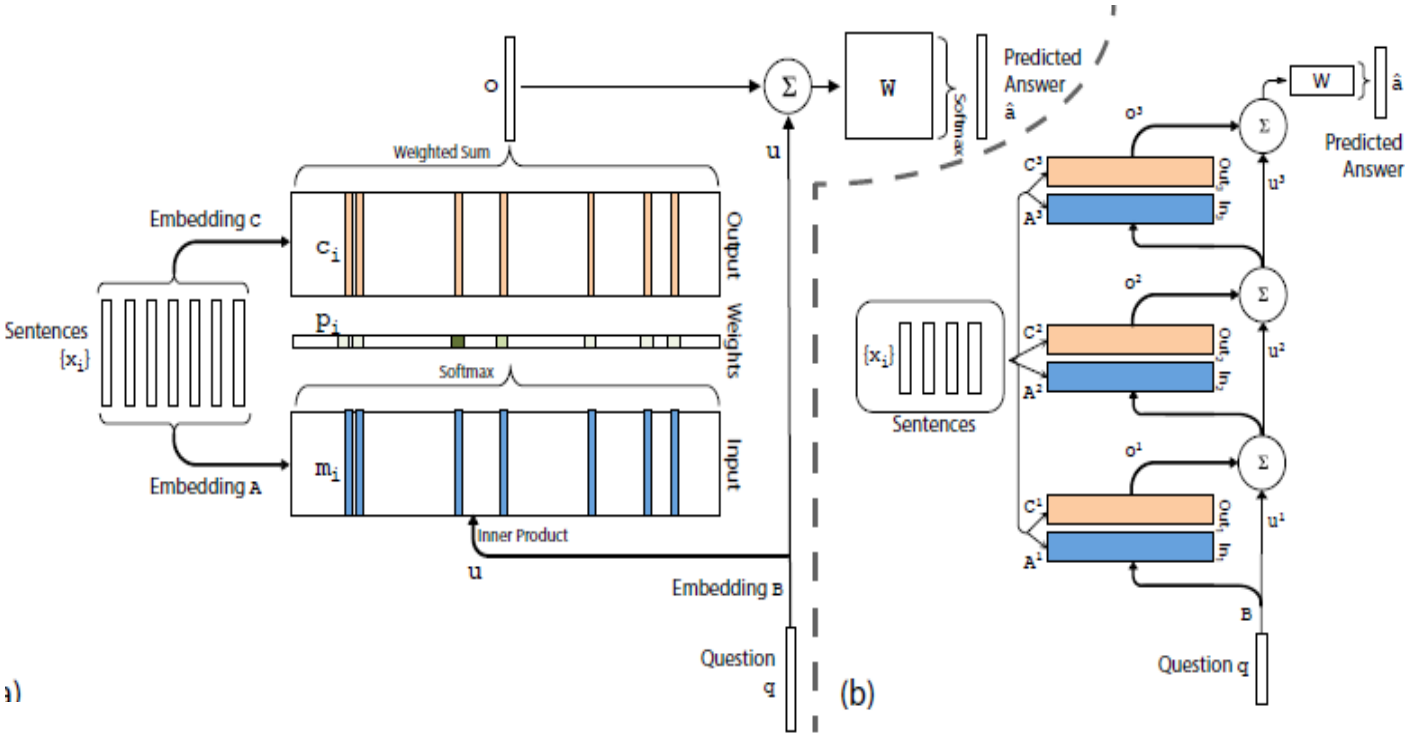
$$m_i = Ax_i$$
$$c_i = Cx_i$$
$$u = Bq$$

- Attention over memories

$$p_i = \text{softmax}(u^T m_i)$$

- Generating (ranking) the final answer

$$o = \sum_i p_i c_i$$
$$a = \text{softmax}(W(o + u))$$



Part II

Deep learning in statistical machine translation (SMT) and Conversation

Tutorial Outline

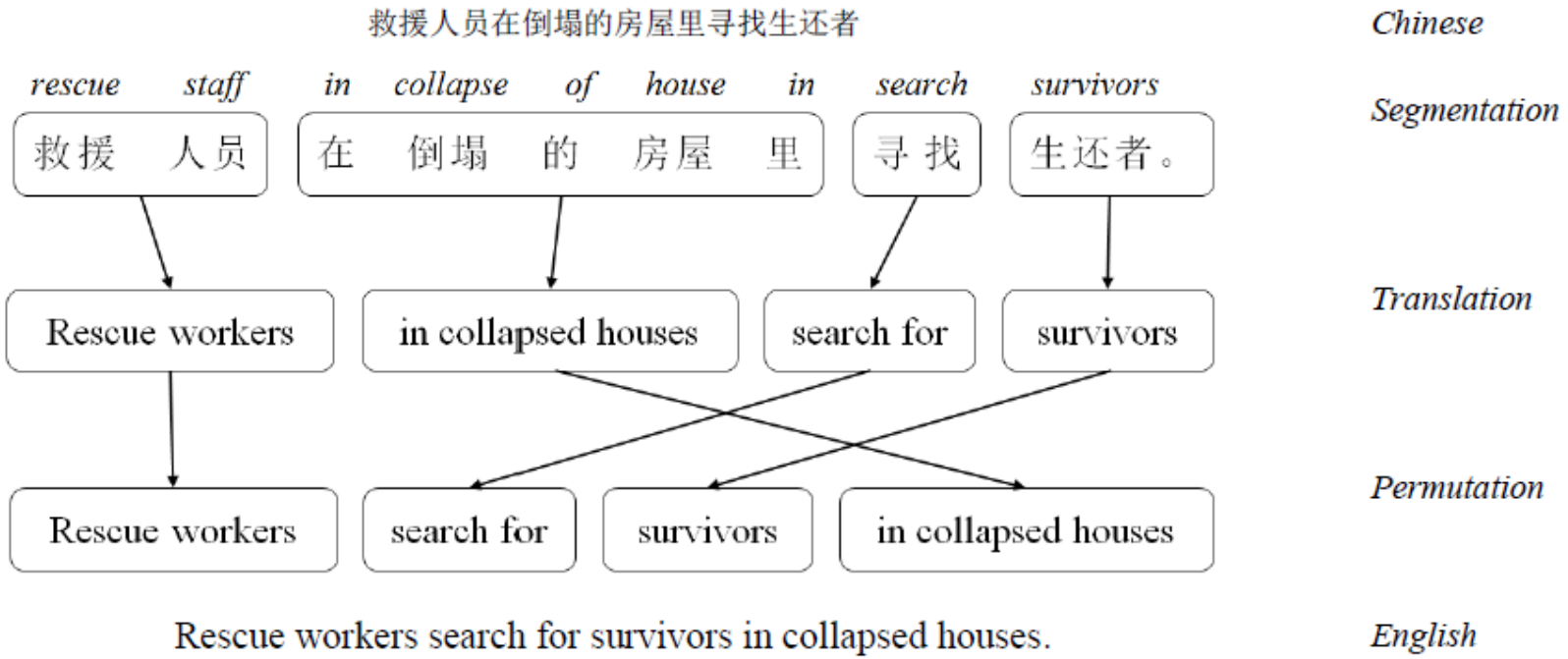
- Part I: Background
- Part II: Deep learning in statistical machine translation (SMT)
  - Review of SMT and DNN in SMT
  - Deep semantic translation models
  - Recurrent neural language models
  - Neural network joint models
  - Neural machine translation
  - Neural conversation models
- Part III: Continuous representations for selected NLP tasks
- Part IV: Natural language understanding
- Part V: Conclusion

Statistical machine translation (SMT)

***S*:** 救援 人员 在 倒塌的 房屋 里 寻找 生还者
***T*:** Rescue workers search for survivors in collapsed houses

- Statistical decision: $T^* = \underset{T}{\operatorname{argmax}}\, P(T|S)$
- Source-channel model: $T^* = \underset{T}{\operatorname{argmax}}\, P(S|T)P(T)$
- Translation models: $P(S|T)$ and $P(T|S)$
- Language model: $P(T)$
- Log-linear model: $P(T|S) = \frac{1}{Z(S,T)} \exp \sum_i \lambda_i h_i(S,T)$
- Evaluation metric: BLEU score (higher is better)

Phrase-based SMT



A taxonomy of neural nets in SMT [Duh 2014]

Core Engine: What is being modeled?

- Target word probability:
  - Language Model: [Schwenk et al., 2012, Vaswani et al., 2013, Niehues and Waibel, 2013, Auli and Gao, 2014]
  - LM w/ Source: [Kalchbrenner and Blunsom, 2013, Auli et al., 2013, Devlin et al., 2014, Cho et al., 2014, Bahdanau et al., 2014, Sundermeyer et al., 2014, Sutskever et al., 2014]
- Translation/Reordering probabilities under Phrase-based MT:
  - Translation: [Maskey and Zhou, 2012, Schwenk, 2012, Liu et al., 2013, Gao et al., 2014a, Lu et al., 2014, Tran et al., 2014, Wu et al., 2014a]
  - Reordering: [Li et al., 2014b]
- Tuple-based MT: [Son et al., 2012, Wu et al., 2014b, Hu et al., 2014]
- ITG Model: [Li et al., 2013, Zhang et al., 2014, Liu et al., 2014]

Related Components:

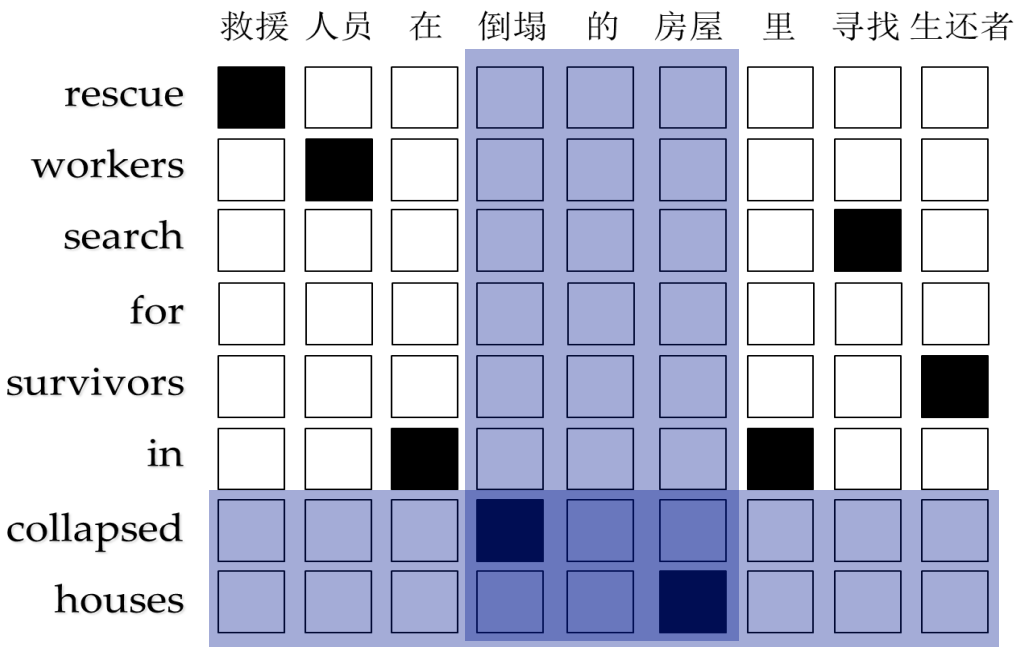
- Word Align: [Yang et al., 2013, Tamura et al., 2014, Songyot and Chiang, 2014]
- Adaptation / Topic Context: [Duh et al., 2013, Cui et al., 2014]
- Multilingual Embeddings: [Klementiev et al., 2012, Lauly et al., 2013, Zou et al., 2013, Kočiský et al., 2014, Faruqui and Dyer, 2014, Hermann and Blunsom, 2014, Chandar et al., 2014]

Examples of NN in phrase-based SMT

- Neural nets as components in log-linear model
  - Translation model $P(T|S)$ or $P(S|T)$: the use of DSSM [Gao+ 14]
  - Language model $P(T)$: the use of RNN [Auli+ 2013; Auli & Gao 14]
  - Joint model $P(t_i|S, t_1 \ldots t_{i-1})$: FFLM + source words [Devlin+ 14]

- Neural machine translation (NMT)
  - Build a single, large NN that reads a sentence and outputs a translation
  - RNN encoder-decoder [Cho+ 2014; Sutskever+ 14]
    - Long short-term memory (gated hidden units)
  - Jointly learning to align and translate [Bahdanau+ 15]
  - NMT surpassed the best result on a WMT task [Luong+ 15]

Phrase translation modeling



(s, t)

(救援, rescue)
(人员, workers)
(在, in)
(倒塌, collapsed)
(房屋, house)
(里, in)
(寻找, search)
(生还者, survivors)
(救援 人员, rescue workers)
(在 倒塌, in collapsed)
(倒塌 的, collapsed)
(的 房屋, house)
(寻找, search for)
(寻找 生还者, search for survivors)
(生还者, for survivors)
(倒塌 的 房屋, collapsed house)

MLE: $P(\boldsymbol{t}|\boldsymbol{s}) = \frac{N(\boldsymbol{s},\boldsymbol{t})}{\sum_{\boldsymbol{t}'} N(\boldsymbol{s},\boldsymbol{t}')}$

Simple, but suffers the data sparseness problem

Deep Semantic Similarity Model (DSSM)

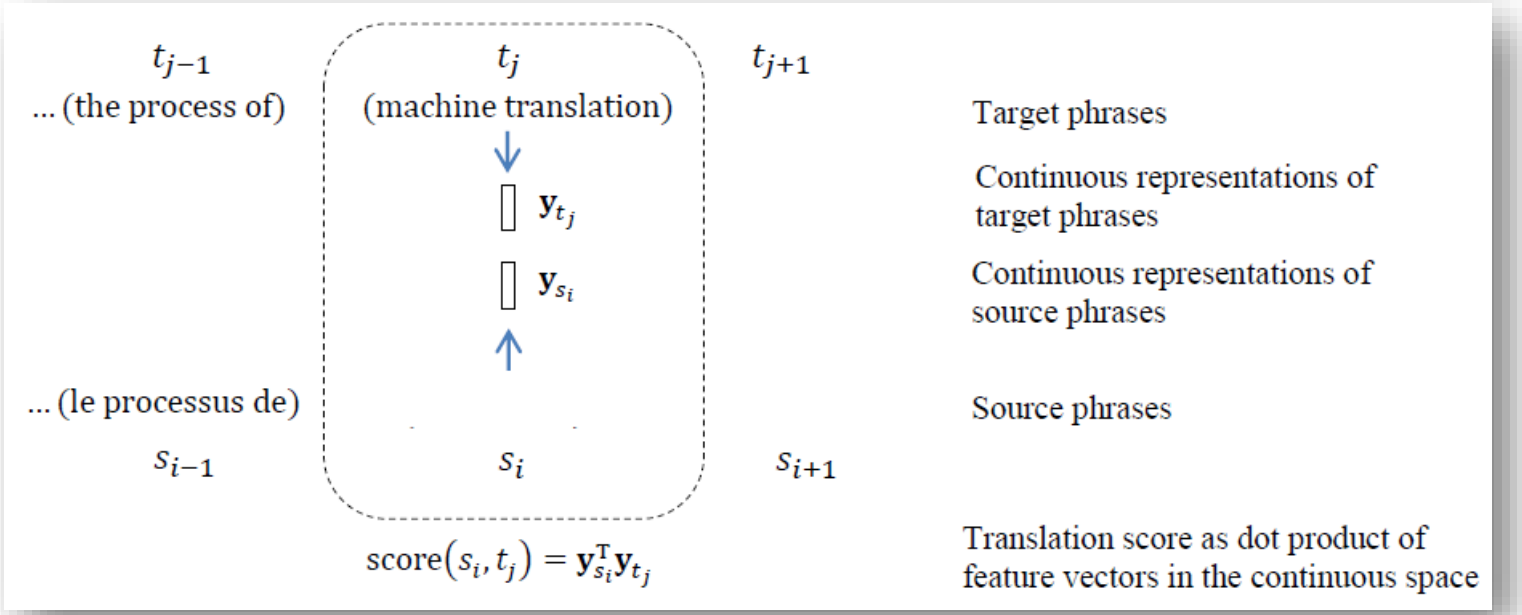
[Huang+ 13; Gao+ 14a; Gao+ 14b; Shen+ 14, Yih+ 15]

- Compute semantic similarity btw text strings X and Y
  - Map X and Y to feature vectors in a latent semantic space via deep neural net
  - Compute the cosine similarity between the feature vectors
  - Also called "Deep Structured Similarity Model" in [Huang+ 13]

- DSSM for NLP tasks

Tasks	X	Y
Machine translation	***Text in language A***	***Translation in language B***
Web search	*Search query*	*Web document*
Image captioning	*Image*	*Caption*
Question Answering	*Question*	*Answer*

DSSM for phrase translation modeling [Gao+ 14a]



- Two neural nets (one for source side, one for target side)
  - Input: bag-of-words representation of source/target phrase
  - Output: vector $\mathbf{y}_s$ for source phrase, $\mathbf{y}_t$ for target phrase
- Phrase translation score = dot product of these vectors
  - $\text{score}(s, t) \equiv \text{sim}_{\boldsymbol{\theta}}(\mathbf{x}_s, \mathbf{x}_t) = \mathbf{y}_s^{\mathrm{T}} \mathbf{y}_t$
- Alleviate data sparsity, enable complex scoring functions, etc.

Model training procedure

- Generate N-best lists using a baseline SMT system
  - Oracle BLEU in N-best is much better than 1-best
- Optimize neural net parameters $\boldsymbol{\theta}$ on the N-best lists of training data
  - Expected BLEU objective: $\mathrm{xBleu}(\boldsymbol{\theta}) = \sum_{T \in \mathrm{GEN}(S_i)} P(T|S_i)\mathrm{sBleu}(T_i, T)$
  - Update $\boldsymbol{\theta}$ with SGD: $\boldsymbol{\theta}^{\boldsymbol{new}} = \boldsymbol{\theta} - \eta \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$,
  - where $\frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \sum_{(s,t)} \frac{\partial \mathcal{L}(\boldsymbol{\theta})}{\partial \mathrm{sim}_{\boldsymbol{\theta}}(\mathbf{x}_s, \mathbf{x}_t)} \frac{\partial \mathrm{sim}_{\boldsymbol{\theta}}(\mathbf{x}_s, \mathbf{x}_t)}{\partial \boldsymbol{\theta}}$
- Incorporate DSSM as a feature in log-linear model
  - Feature weight is optimized using MERT on development data.
  - No decoder modification
- Loop if desired

N-gram language modeling

- Word n-gram model (e.g., n = 3)
  - A word depends only on n-1 preceding words
  - $P(w_1 w_2 \dots w_n) = P(w_1)P(w_2|w1) \prod_{i=2\dots n} P(w_i|w_{i-2}w_{i-1})$
  - Cannot capture long-distance dependency

- Problem of using long history
  - Rare events: unreliable probability estimates

model		**# parameters**
unigram	$P(w_1)$	20,000
bigram	$P(w_2\|w_1)$	400M
trigram	$P(w_3\|w_1w_2)$	8×10^{12}
4-gram	$P(w_4\|w_1w_2w_3)$	1.6×10^{17}

[Manning & Schütze 99]

Recurrent neural net for language modeling

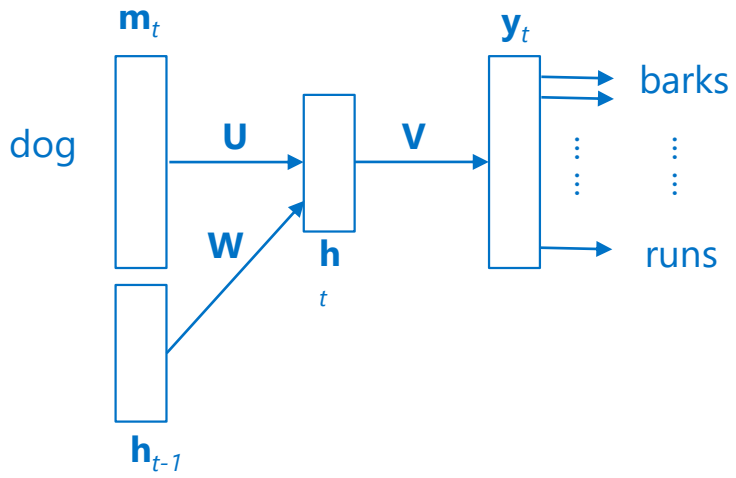


Table 1: *Performance of models on WSJ DEV set when increasing size of training data.*

Model	# words	PPL	WER
KN5 LM	200K	336	16.4
KN5 LM + RNN 90/2	200K	271	15.4
KN5 LM	1M	287	15.1
KN5 LM + RNN 90/2	1M	225	14.0
KN5 LM	6.4M	221	13.5
KN5 LM + RNN 250/5	6.4M	156	11.7

[Mikolov+ 11]

m_t: input one-hot vector at time step t
h_t: encodes the history of all words up to time step t
y_t: distribution of output words at time step t

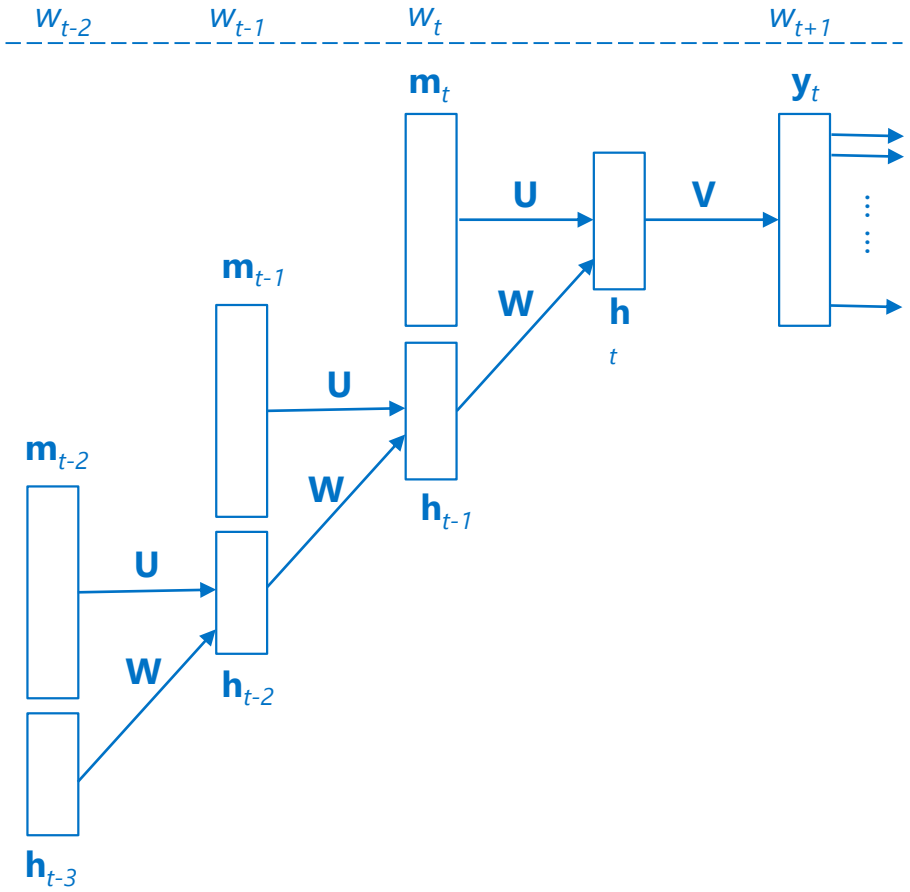
$$\mathbf{z}_t = \mathbf{U}\mathbf{m}_t + \mathbf{W}\mathbf{h}_{t-1}$$
$$\mathbf{h}_t = \sigma(\mathbf{z}_t)$$
$$\mathbf{y}_t = g(\mathbf{V}\mathbf{h}_t)$$

where

$$\sigma(z) = \frac{1}{1+\exp(-z)}, \quad g(z_m) = \frac{\exp(z_m)}{\sum_k \exp(z_k)}$$

$g(.)$ is called the *softmax* function

RNN unfolds into a DNN over time



$$\mathbf{z}_t = \mathbf{U}\mathbf{m}_t + \mathbf{W}\mathbf{h}_{t-1}$$
$$\mathbf{h}_t = \sigma(\mathbf{z}_t)$$
$$\mathbf{y}_t = g(\mathbf{V}\mathbf{h}_t)$$

where

$$\sigma(z) = \frac{1}{1+\exp(-z)}, \quad g(z_m) = \frac{\exp(z_m)}{\sum_k \exp(z_k)}$$

RNN LM decoder integration [Auli & Gao 14]

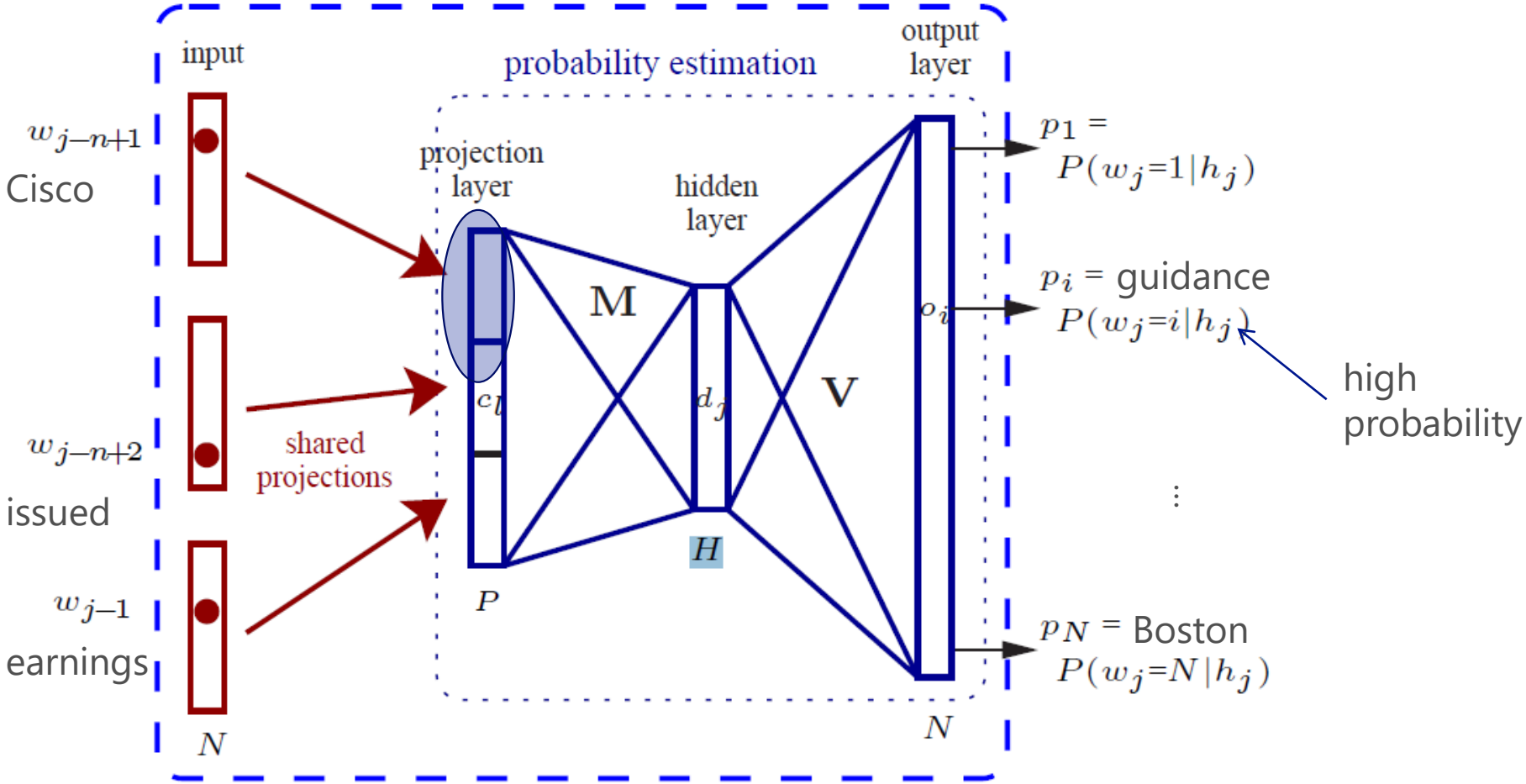
- RNN LMs require history going back to start-of-sentence. Harder to do dynamic programming.
- To score new words, each decoder state needs to maintain *h*. For recombination, merge hypotheses by traditional n-gram context and the best *h*

	WMT12 Fr-En	WMT12 De-En
baseline (n-gram)	24.85	19.80
100-best rescoring	25.74	20.54
lattice rescoring	26.43	20.63
decoding	26.86	20.93

Joint model: language model with source

- $P(t_i|t_{i-2}t_{i-1}, S)$
- How to model S?
  - Entire source sentence or aligned source words
  - S as a word sequence, bag of words, or vector representation
  - How to learn the vector representation of S?
- Neural network joint models based on
  - RNN language model [Auli+ 13]
  - Feedforward neural language model [Devlin+ 14]

Feed-forward neural language model [Bengio+ 03]



Joint model of [Devlin+ 14]

- Extend feed-forward LM to include window around aligned source words.
  - Heuristic: if align to multiple source words, choose middle; if unaligned, inherit alignment from closest target word
- Train on bitext with alignment; optimize target likelihood.

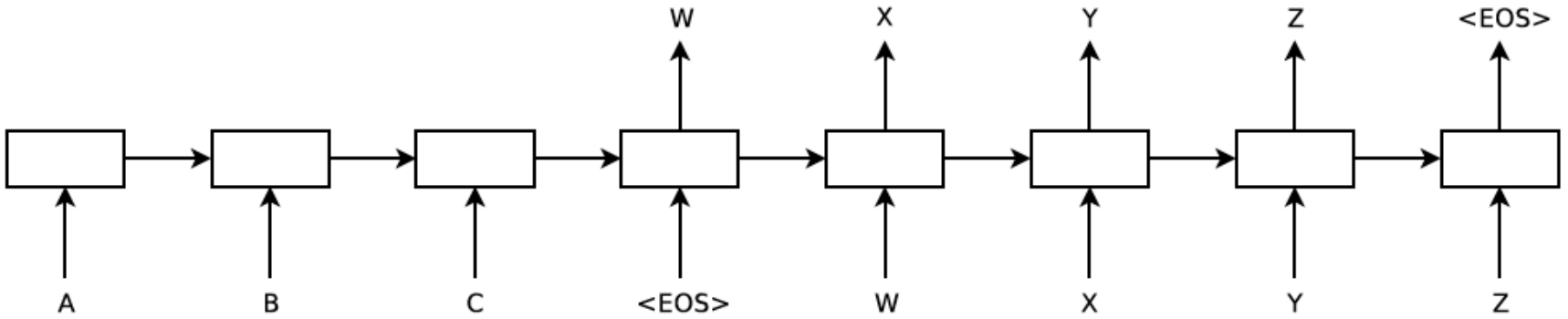
Neural machine translation

[Sutskever+ 14; Cho+ 14; Bahdanau+ 15; Luong+ 15]

- Build a single, large NN that reads a sentence and outputs a translation
  - Unlike phrase-based system that consists of many component models
- Encoder-decoder based approach
  - An encoder RNN reads and encodes a source sentence into a fixed-length memory vector
  - A decoder RNN outputs a variable-length translation from the encoded memory vector
  - Encoder-decoder RNNs are jointly learned on bitext, optimizing target likelihood

Encoder-decoder model of [Sutskever+ 2014]

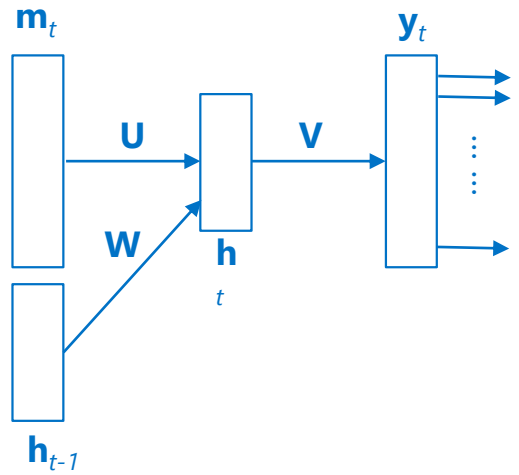
- "A B C" is source sentence; "W X Y Z" is target sentence

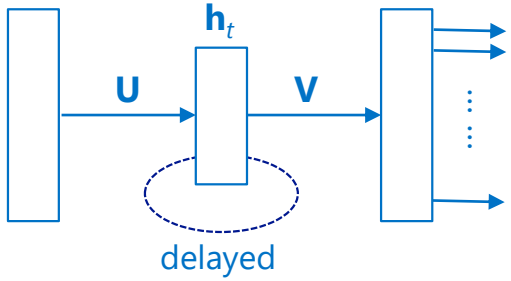


- Treat MT as general sequence-to-sequence transduction
  - Read source; accumulate hidden state; generate target
  - <EOS> token stops the recurrent process
  - In practice, read source sentence in reverse leads to better MT results
- Train on bitext; optimize target likelihood using SGD

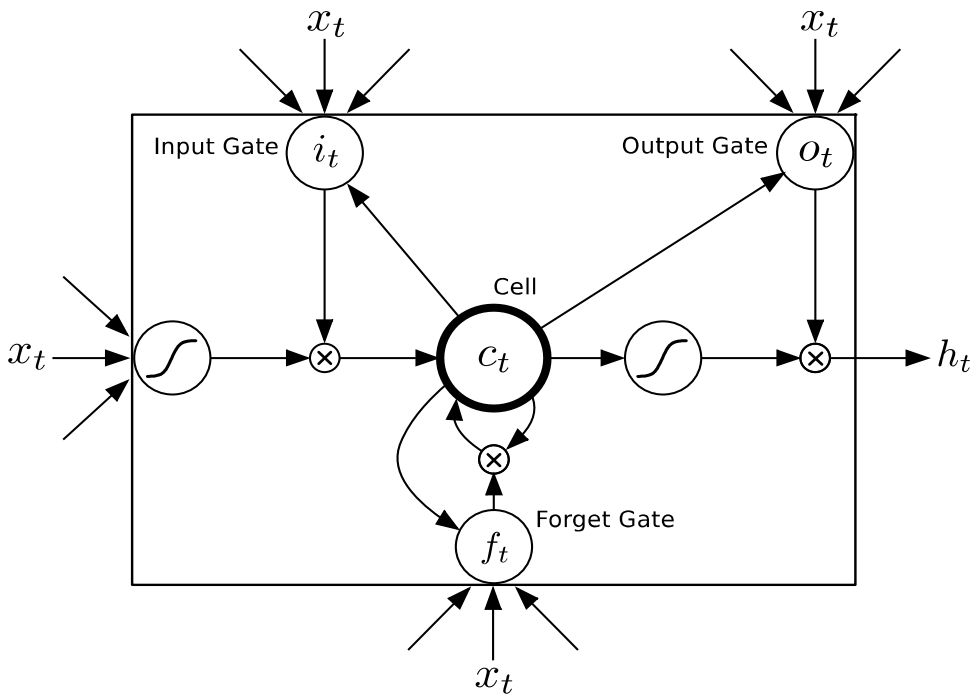
Potentials and difficulties of RNN

- In theory, RNN can "store" in $\boldsymbol{h}$ all information about past inputs
- But in practice, standard RNN cannot capture very long distance dependency
  - Vanishing/exploding gradient problem in backpropagation
  - Not robust to noise
- Solution: long short-term memory (LSTM)





A long short-term memory cell
[Hochreiter & Schmidhuber 97; Graves+ 13]



$$i_t = \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right)$$
$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right)$$
$$c_t = f_t c_{t-1} + i_t \tanh\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right)$$
$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right)$$
$$h_t = o_t \tanh(c_t)$$

Information flow in an LSTM unit of the RNN, with both diagrammatic and mathematical descriptions. W's are weight matrices, not shown but can easily be inferred in the diagram (Graves et al., 2013).

A 2-gate memory cell [Cho+ 14]

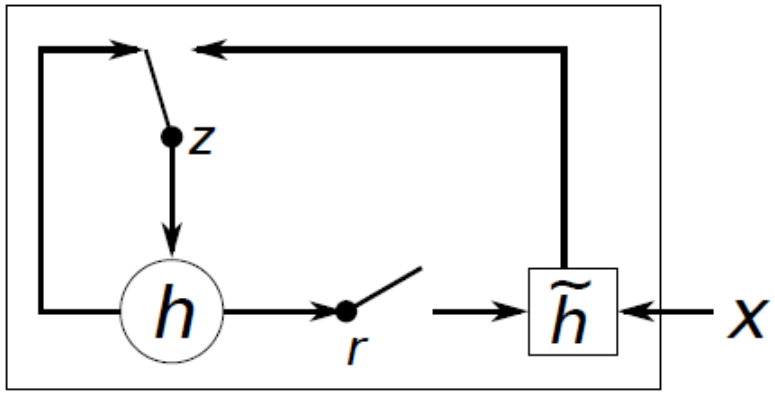


Figure 2: An illustration of the proposed hidden activation function. The update gate z selects whether the hidden state is to be updated with a new hidden state $\tilde{h}$. The reset gate r decides whether the previous hidden state is ignored. See

$$r_j = \sigma\left(\left[\mathbf{W}_r\mathbf{x}\right]_j + \left[\mathbf{U}_r\mathbf{h}_{\langle t-1\rangle}\right]_j\right)$$

$$z_j = \sigma\left(\left[\mathbf{W}_z\mathbf{x}\right]_j + \left[\mathbf{U}_z\mathbf{h}_{\langle t-1\rangle}\right]_j\right)$$

$$\tilde{h}_j^{\langle t\rangle} = \phi\left(\left[\mathbf{W}\mathbf{x}\right]_j + \left[\mathbf{U}\left(\mathbf{r}\odot\mathbf{h}_{\langle t-1\rangle}\right)\right]_j\right)$$

$$h_j^{\langle t\rangle} = z_j h_j^{\langle t-1\rangle} + (1-z_j)\tilde{h}_j^{\langle t\rangle}$$

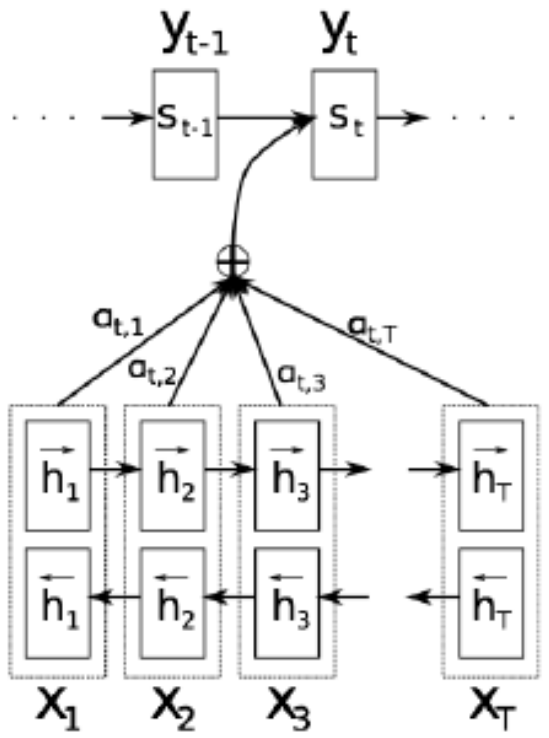
Joint learning to align and translate

- Issue with encoder-decoder model for SMT
  - Compressing a source sentence into a fixed-length vector makes it difficult for RNN to cope with long sentences.

- Attention model of [Bahdanan+ 15]
  - Encodes the input sentence into a sequence of vectors and choose a subset of these vectors adaptively while decoding
  - An idea similar to that of [Devlin+ 14]

Attention model of [Bahdanan+ 15]

- Encoder:
  - bidirectional RNN to encode each word and its context
- Decoder:
  - Searches for a set of source words that are most relevant to the target word to be predicted.
  - Predicts a target word based on the context vectors associated with these source words and all the previous generated target words.
- Close to state-of-the-art performance
  - **Better at translating long sentences**



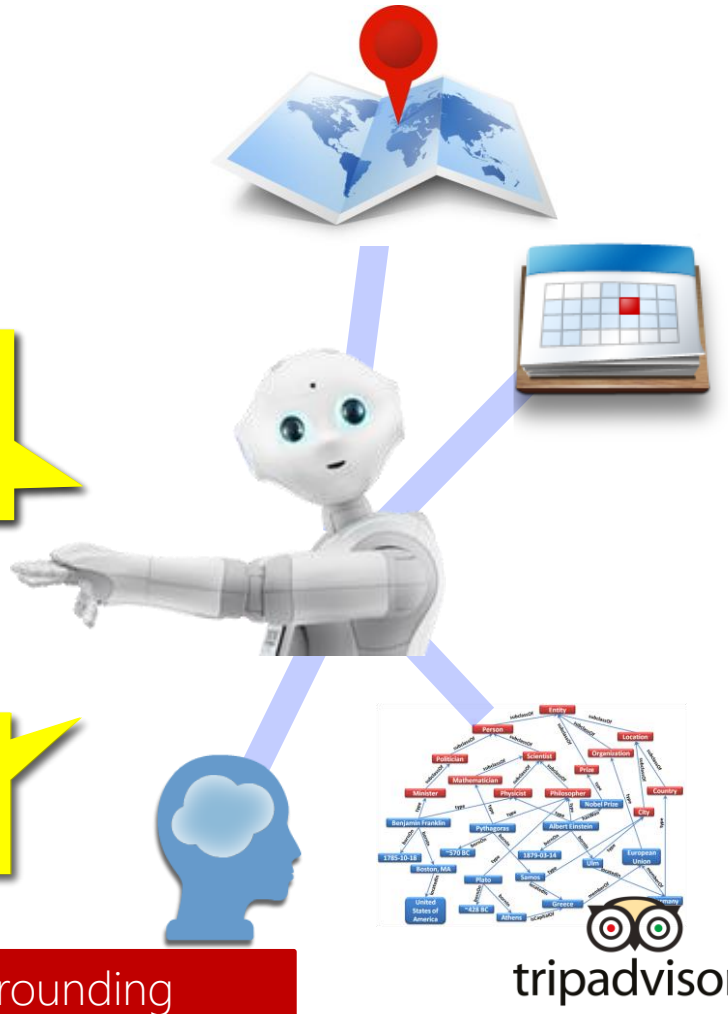
Enable people to *converse* with their devices -- an E2E neural approach

The weather is so depressing these days.

I know, I dislike rain too. What about spending **the weekend in eastern Washington**?

Any recommendation?

Try **Dry Falls**, it's **spectacular**! There's a **cheap hotel** nearby - **Rodeway Inn**.



Need grounding in the real world!

- Natural, human-like interaction
- Decision support

Slides from Michel Galley

MSR's Neural Conversation Engine based on Seq2Seq Models [Sordoni+ 15; Li+ NAACL16; Li+ ACL16]

Neural model learns the general shape of conversation, and the system output is situationally appropriate and coherent…

H: hi
M: how are you ?
H: not bad
M: what 's wrong ?
H: nothing really
M: wan na talk about it ? i 'm here if you wan na talk x
H: well , i 'm just tired
M: me too , i 'm here if you wan na talk
H: thanks , what are you doing ?
M: you 're welcome and i 'm watching tv and you ?
H: on netflix ?
M: yeah , i 'm watching it on netflix .
H: what are you watching ?
M: a tv show .
H: nice , did you have dinner already ?
M: no , not yet . what about you ?

A more goal-oriented example. No grounding into a real calendar, but the "shape" of the conversation is fluent and plausible…

H: would thursday afternoon work sometime ?
M: yeah , sure . just let me know when you 're free .
H: after lunch is probably the best time
M: okay , sounds good . just let me know when you 're free .
H: would 2 pm work for you ?
M: works for me .
H: well let 's say 2 pm then i 'll see you there
M: sounds good .

- Models are trained Twitter conversation dataset
- Generating responses vs. retrieving responses
- Easy to incorporate contextual info via embedding
  - User profile – personalized conversation
  - knowledge bases – grounded conversation
- The engine is E2E learned from conversation experience
  - Learning a goal-oriented conversation engine via RL

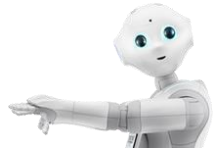
Neural Response Generation: The Blandness Problem

How was your weekend?

I don't know.

What did you do?

I don't understand what you are talking about.



This is getting boring…

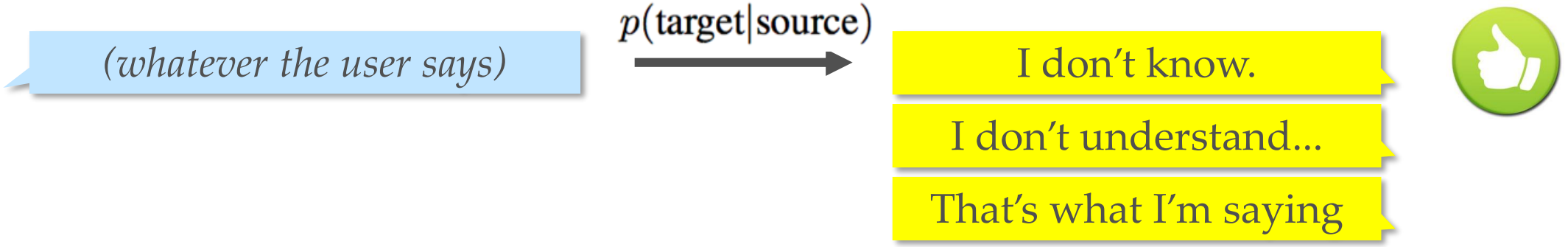
Yes that's what I'm saying.

Slides from Michel Galley

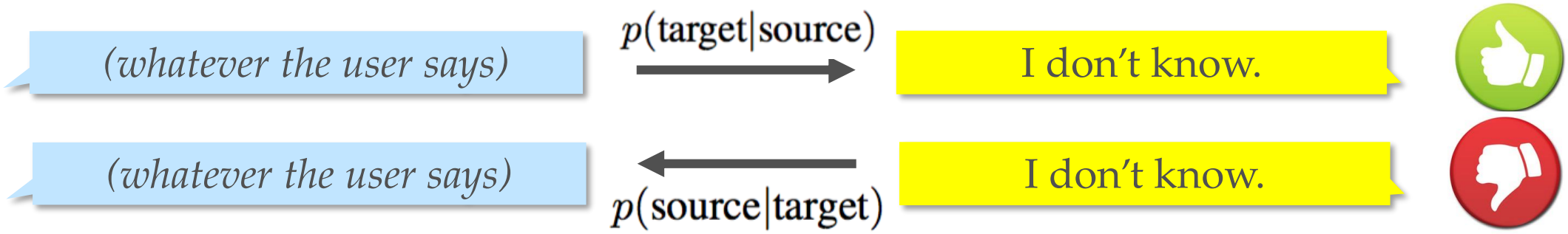
Blandness problem: cause and remedies

[Li et al., NAACL 2016]

Common ML objective (maximum likelihood)



Mutual information objective:



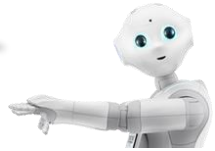
Beyond blandness: Examples

Wow sour starbursts really do make **your mouth water**... mm drool.
Can I have one?

Of course you can! They're **delicious**!

Milan apparently **selling Zlatan** to balance the books... **Where next**, Madrid?

I think he'd be a **good signing**.



'tis a fine **brew** on a day like this! Strong though, **how many** is sensible?

Depends on how much you **drink**!

Well he was on in Bromley a while ago... **still touring**.

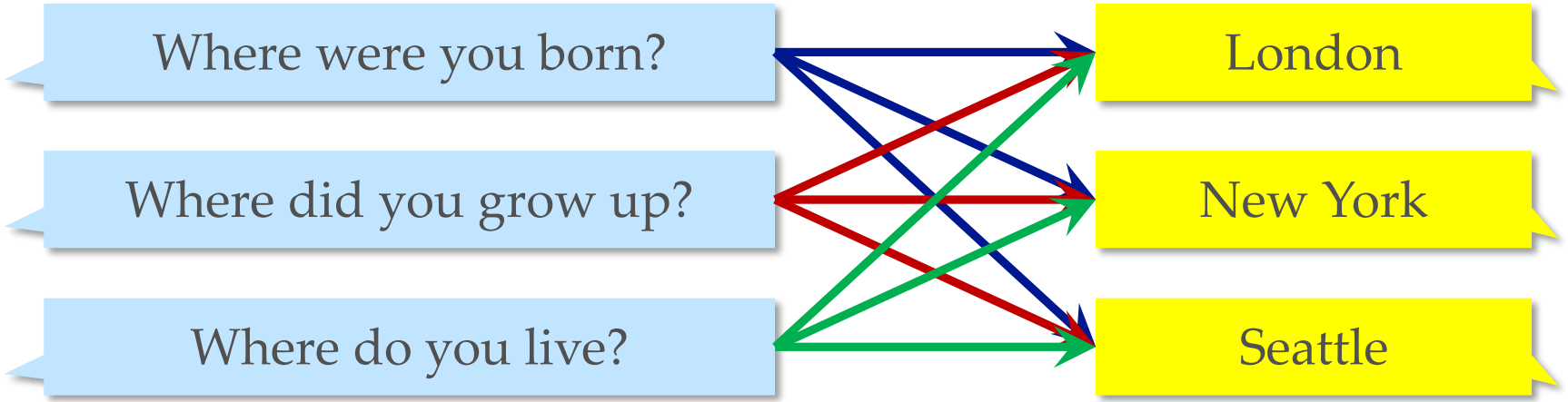
I've never **seen him live**.

A Persona-Based Conversational Model

Why? Motivation is to model:

- personal background
- behavioral and stylistic differences (e.g., introvert vs. extrovert)

Better at "explaining away" conversational data:



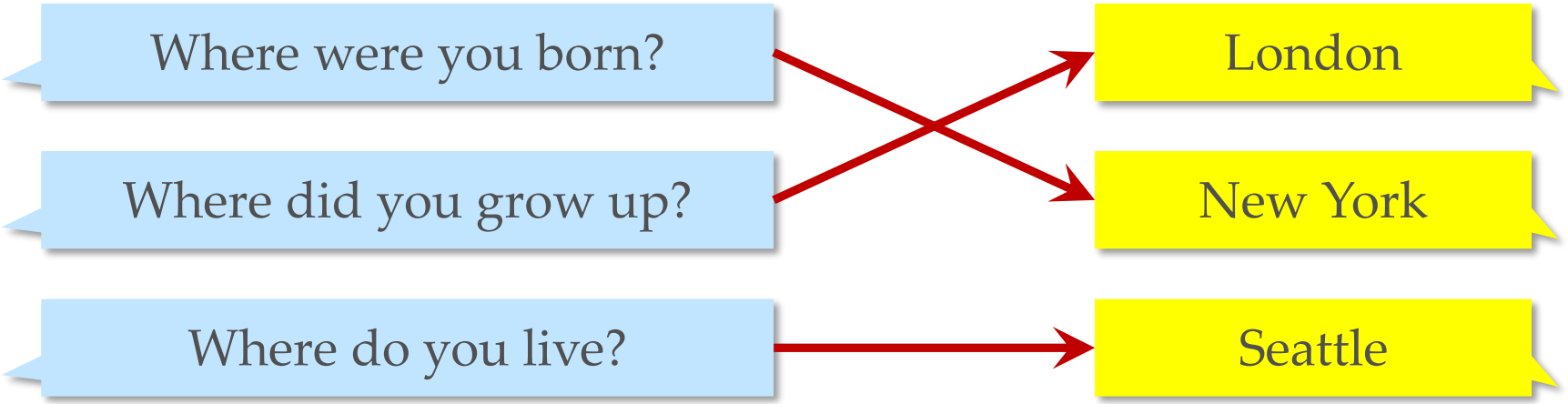
Conversation data is badly entangled! (N-to-1, 1-to-N)

A Persona-Based Conversational Model

Why? Motivation is to model:

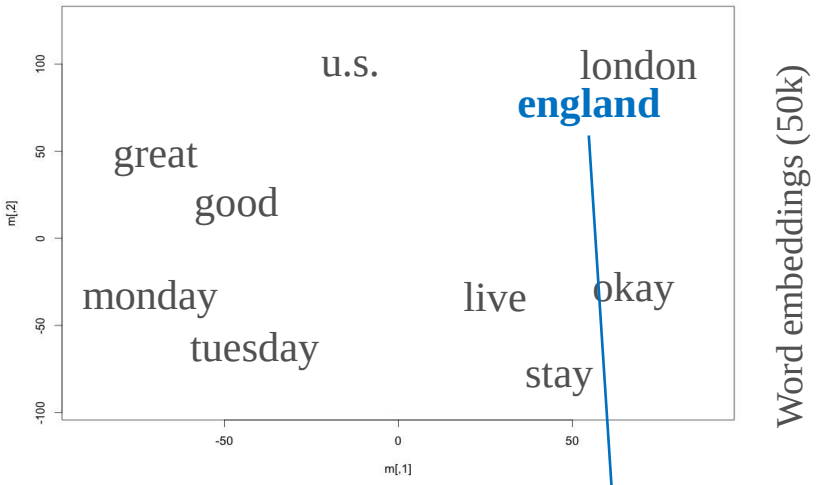
- personal background
- behavioral and stylistic differences (e.g., introvert vs. extrovert)

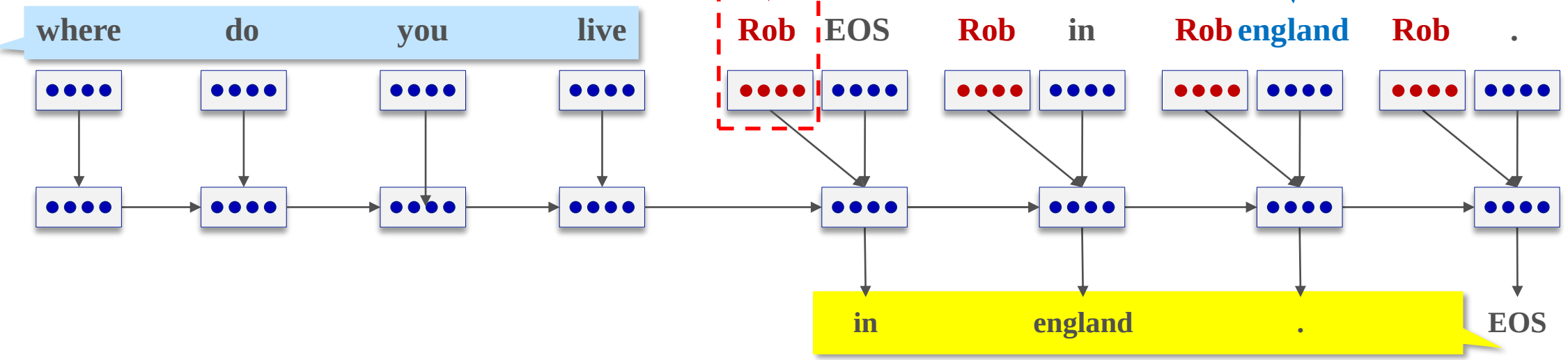
Better at "explaining away" conversational data:



$p(T|S, \text{PERSON\_943})$

Personalized Response Generation





Sample outputs

User1	
message	Where is your hometown?
response	I'm from England.
message	Where are you from?
response	I'm from England.
message	In which city do you live now?
response	I live in London.
message	In which country do you live now?
response	I live in England.
message	What is your major?
response	Business. You?
message	What did you study in college?
response	I did business studies.
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 18.

User2	
message	Where is your hometown?
response	I live in Indonesia.
message	Where are you from?
response	England, you?
message	In which city do you live now?
response	I live in Jakarta.
message	In which country do you live now?
response	I live in Indonesia.
message	What is your major?
response	Business, you?
message	What did you study in college?
response	Psychology, you?
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 16.

Sample outputs

User1	
message	Where is your hometown?
response	I'm from England.
message	Where are you from?
response	I'm from England.
message	In which city do you live now?
response	I live in London.
message	In which country do you live now?
response	I live in England.
message	What is your major?
response	Business. You?
message	What did you study in college?
response	I did business studies.
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 18.

User2	
message	Where is your hometown?
response	I live in Indonesia.
message	Where are you from?
response	England, you?
message	In which city do you live now?
response	I live in Jakarta.
message	In which country do you live now?
response	I live in Indonesia.
message	What is your major?
response	Business, you?
message	What did you study in college?
response	Psychology, you?
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 16.

Sample outputs

User1	
message	Where is your hometown?
response	I'm from England.
message	Where are you from?
response	I'm from England.
message	In which city do you live now?
response	I live in London.
message	In which country do you live now?
response	I live in England.
message	What is your major?
response	Business. You?
message	What did you study in college?
response	I did business studies.
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 18.

User2	
message	Where is your hometown?
response	I live in Indonesia.
message	Where are you from?
response	England, you?
message	In which city do you live now?
response	I live in Jakarta.
message	In which country do you live now?
response	I live in Indonesia.
message	What is your major?
response	Business, you?
message	What did you study in college?
response	Psychology, you?
message	How old are you?
response	I'm 18.
message	What is your age?
response	I'm 16.

Interim summary

- Part I: Background
  - A brief history of deep neural networks (DNN)
  - An example of neural models for query classification
  - Different forms of DNN for classification/ranking/generation tasks
- Part II: Deep learning in statistical machine translation and conversation
  - Review of SMT and DNN in SMT
  - Deep semantic translation models
  - (Recurrent neural language models)
  - (Neural network joint models)
  - Neural machine translation (Seq2Seq models)
  - Neural conversation models (Seq2Seq models)
- Part III: Continuous representations for selected NLP tasks
- Part IV: Natural language understanding
- Part V: Conclusion

Part III

Continuous representations for selected NLP tasks

Continuous representations for selected NLP tasks

- Deep semantic similarity model (DSSM) for information retrieval & entity ranking
- Deep reinforcement learning in a continuous semantic space for NLP
- Multimodal semantic learning & inference for image captioning and visual question answering

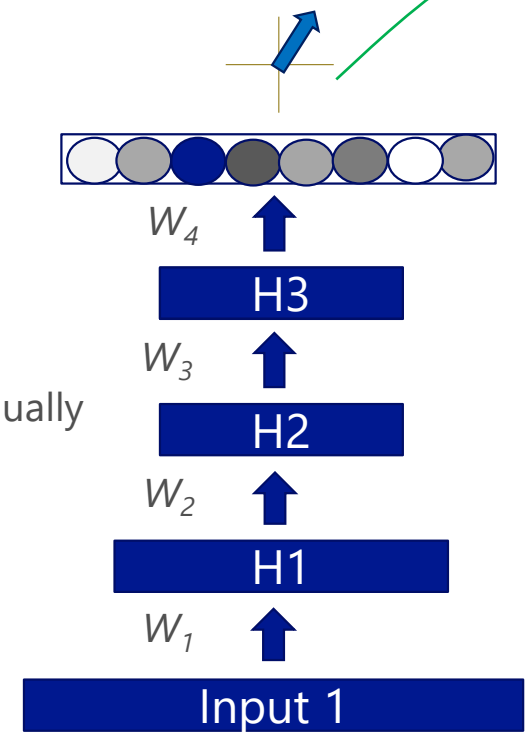
Learning continuous semantic representations for natural language

e.g., from a raw sentence to an abstract semantic vector (Sent2Vec)

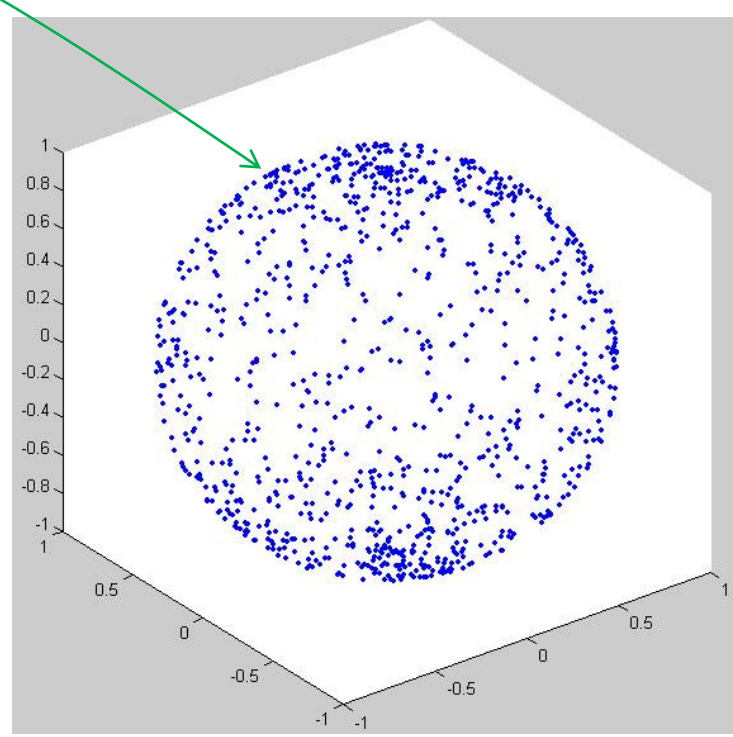
Abstract representation in the semantic space

each non-linear layer gradually extracts deeper invariance

Raw text, e.g., a sequence of words



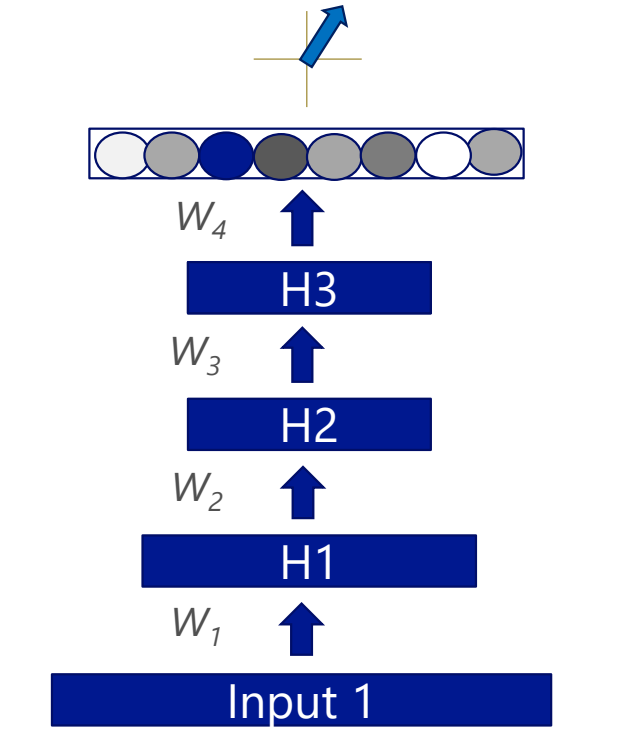
a man is reading the new york times



Sent2Vec is crucial in many NLP tasks

Tasks	***Source***	***Target***
Web search	*search query*	*web documents*
Ad selection	*search query*	*ad keywords*
Contextual entity ranking	*mention (highlighted)*	*entities*
Online recommendation	*doc in reading*	*interesting things / other docs*
Machine translation	*phrases in language S*	*phrases in language T*
Knowledge-base construction	*entity*	*entity*
Question answering	*pattern | mention*	*relation | entity*
Personalized recommendation	*user*	*app, movie, etc.*
Image search	*query*	*image*
Image captioning	*image*	*text*
…		

The supervision problem:



a man is reading the new york times

However

- the semantic meaning of texts – to be learned – is latent
- no clear target for the model to learn
- How to do back-propagation?

Fortunately

- we usually know if two texts are "similar" or not.
- That's the signal for semantic representation learning.

Deep Structured Semantic Model

Deep Structured Semantic Model/Deep Semantic Similarity Model (**DSSM**)

project the whole sentence to a continuous semantic space – e.g., *Sentence to Vector*.

The DSSM is built upon **characters** (rather than words) for scalability and generalizability

The DSSM is trained by optimizing an **similarity-driven** objective

Huang, He, Gao, Deng, Acero, Heck, "Learning deep structured semantic models for web search using clickthrough data," CIKM, October, 2013

Character-level coding (a.k.a. word hashing)

- E.g., character-trigram based *Word Hashing* of "cat"
  - -> #cat#
  - Tri-characters: #-c-a, c-a-t, a-t-#.
- Compact representation
  - |Voc| (500K) → |Char-trigram| (30K)
- Generalize to unseen words
- Robust to misspelling, inflection, etc.

$$x\,(cat) = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$ ← The index of word *cat* in the vocabulary

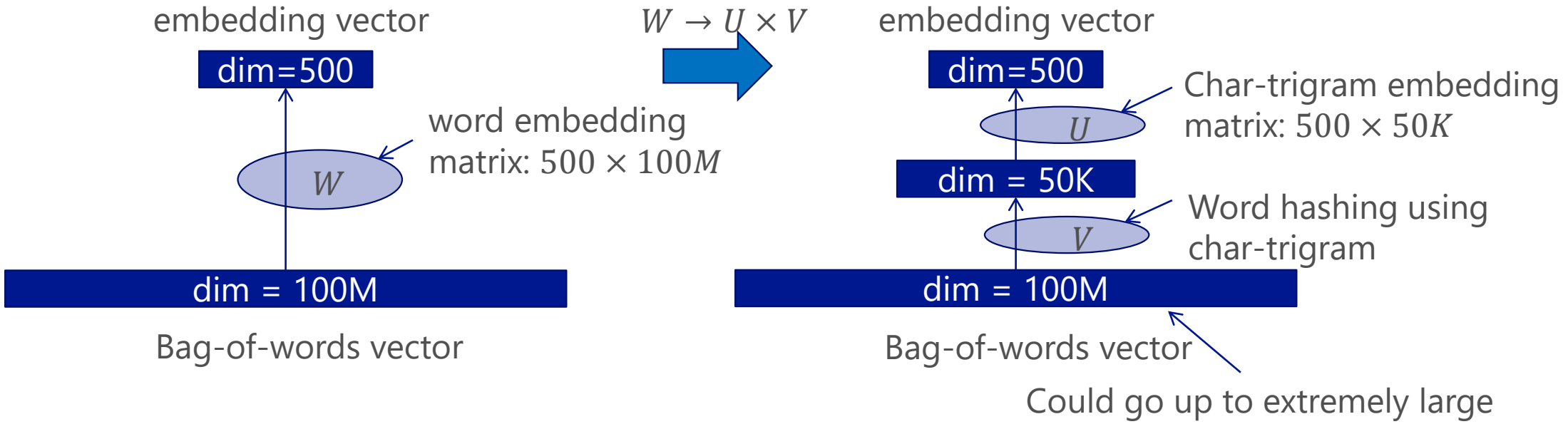
$$f(cat) = \begin{bmatrix} \vdots \\ 1 \\ \vdots \\ 1 \\ 1 \\ \vdots \end{bmatrix}$$ → Indices of *#-c-a, c-a-t, a-t-#* in the letter-tri-gram list, respectively.

What if different words have the same word hashing code (collision)?

Vocabulary size	Unique letter-tg observed in voc	Number of Collisions
40K	10306	2 (0.005%)
500K	30621	22 (0.004%)

DSSM: built at the character-level

Decompose *any* word into set of context-dependent characters



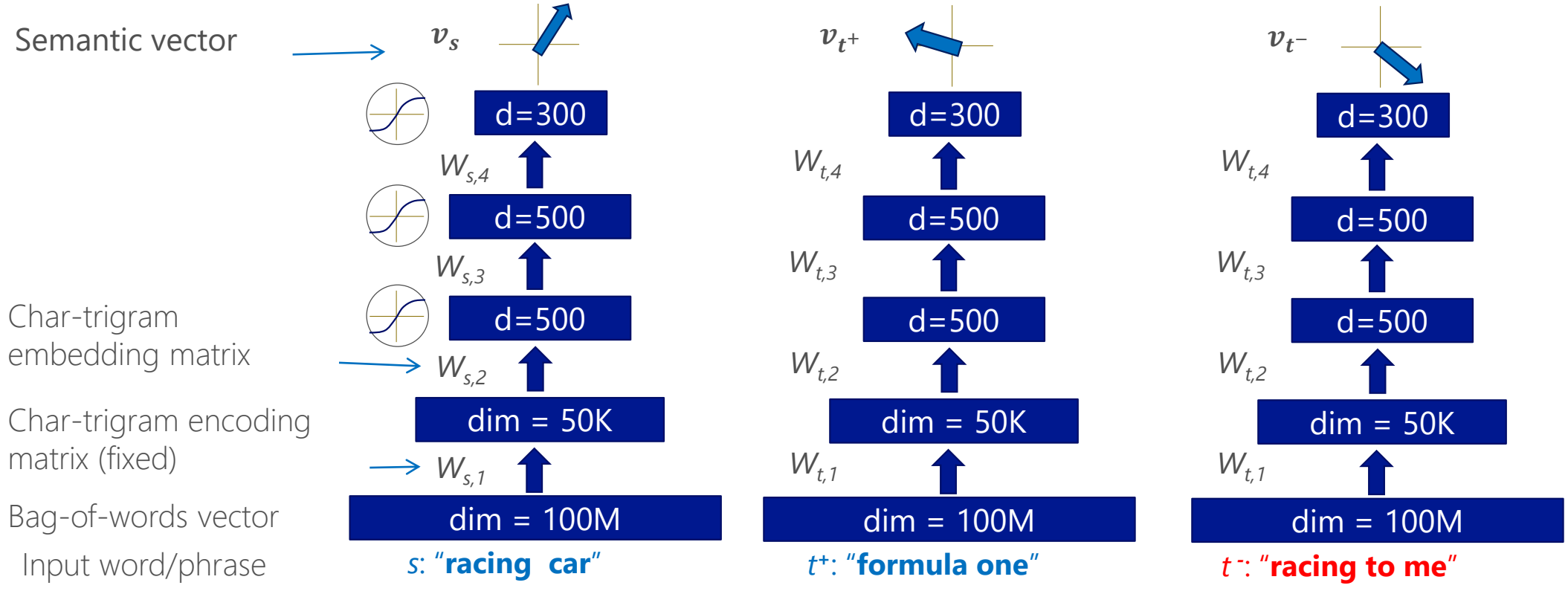
Preferable for large scale NL tasks

- Arbitrary size of vocabulary (*scalability*)
- Misspellings, word fragments, new words, etc. (*generalizability*)

DSSM: a similarity-driven Sent2Vec model

Initialization:

Neural networks are initialized with random weights

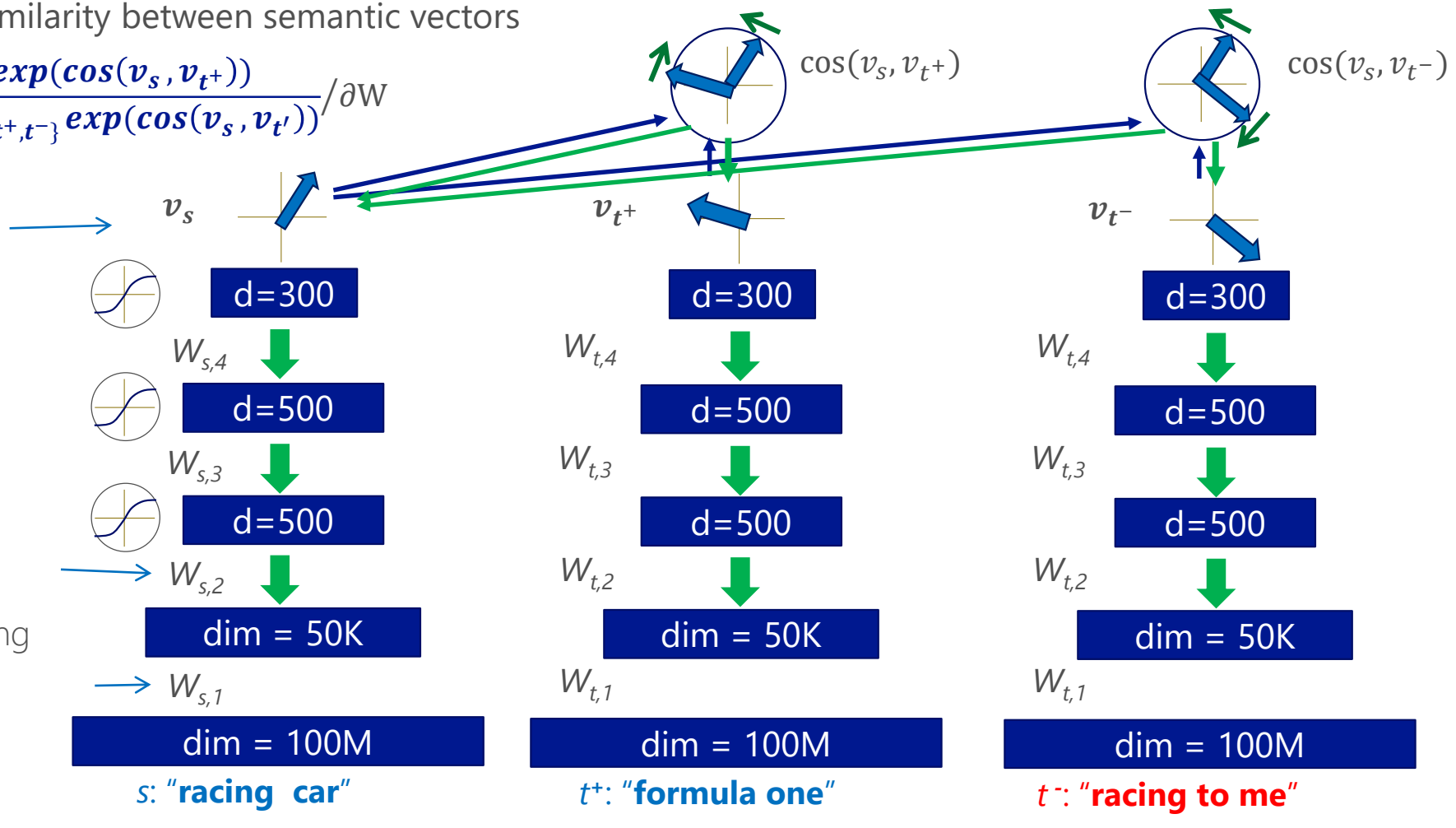


DSSM: a similarity-driven Sent2Vec model

Training:

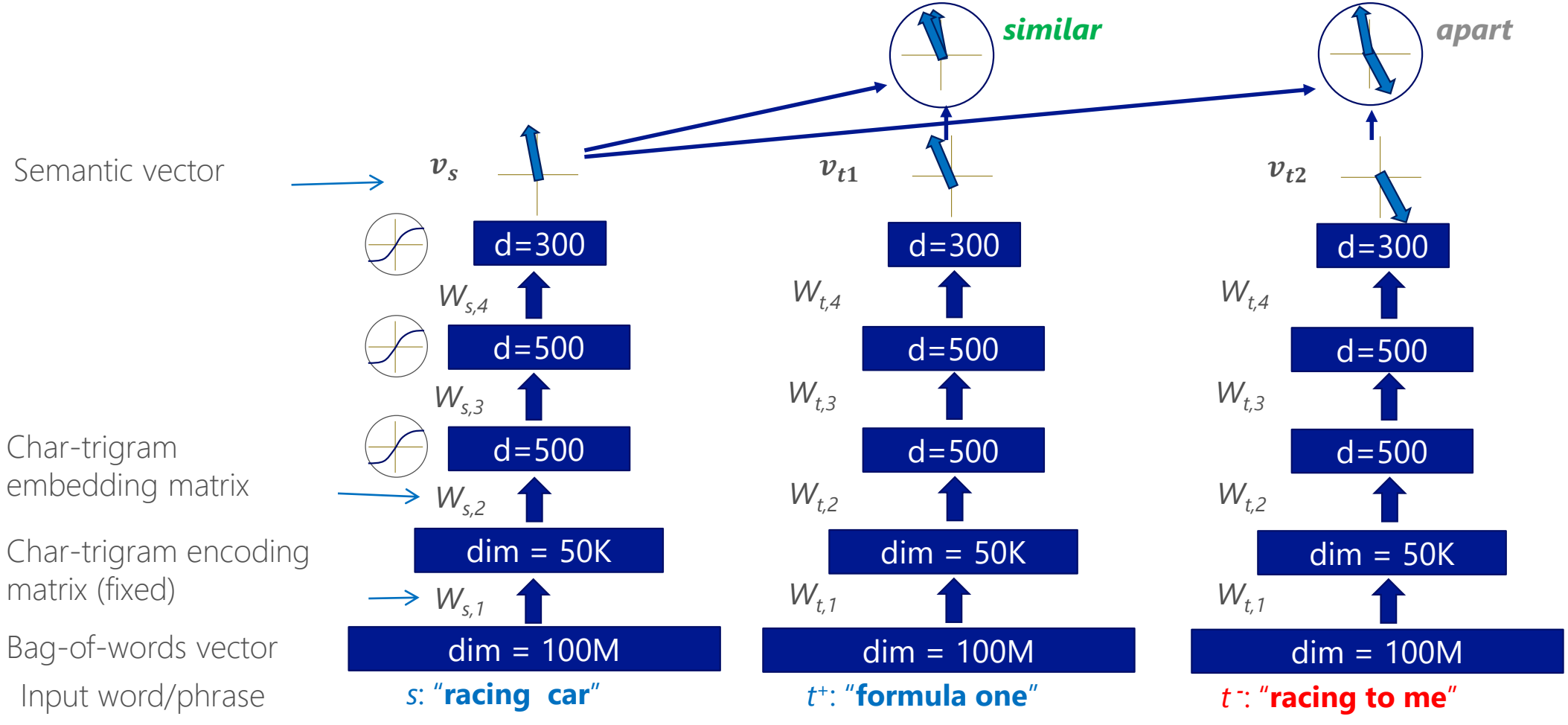
Compute Cosine similarity between semantic vectors

Compute gradients $\partial \frac{\boldsymbol{exp(cos(v_s, v_{t^+}))}}{\sum_{t'=\{t^+,t^-\}} \boldsymbol{exp(cos(v_s, v_{t'}))}} / \partial W$



DSSM: a similarity-driven Sent2Vec model

Runtime:



Training objectives

Objective: cosine similarity based loss

Using web search as an example:

- a query q and a list of docs $D = \{d^+, d_1^-, \dots d_K^-\}$
  - d^+ positive doc; $d_1^-, \dots d_K^-$ are negative docs to q (e.g., sampled from not clicked docs)
- Objective: the posterior probability of the clicked doc given the query

$$P_\theta(d^+|q) = \frac{\exp\left(\gamma\, cos(v_{\boldsymbol{\theta}}(q), v_{\boldsymbol{\theta}}(d^+))\right)}{\sum_{d \in \boldsymbol{D}} \exp\left(\gamma\, cos(v_{\boldsymbol{\theta}}(q), v_{\boldsymbol{\theta}}(d))\right)}$$

e.g., $v_{\boldsymbol{\theta}}(q) = \sigma(W_{s,4} \times \sigma(W_{s,3} \times \sigma(W_{s,2} \times ltg(q))))$

$v_{\boldsymbol{\theta}}(d) = \sigma(W_{t,4} \times \sigma(W_{t,3} \times \sigma(W_{t,2} \times ltg(d))))$

where $\theta = \{W_{s,2\sim4}, W_{t,2\sim4}\}, \sigma(\,)$ is a tanh function.

Using Convolutional Neural Net in DSSM

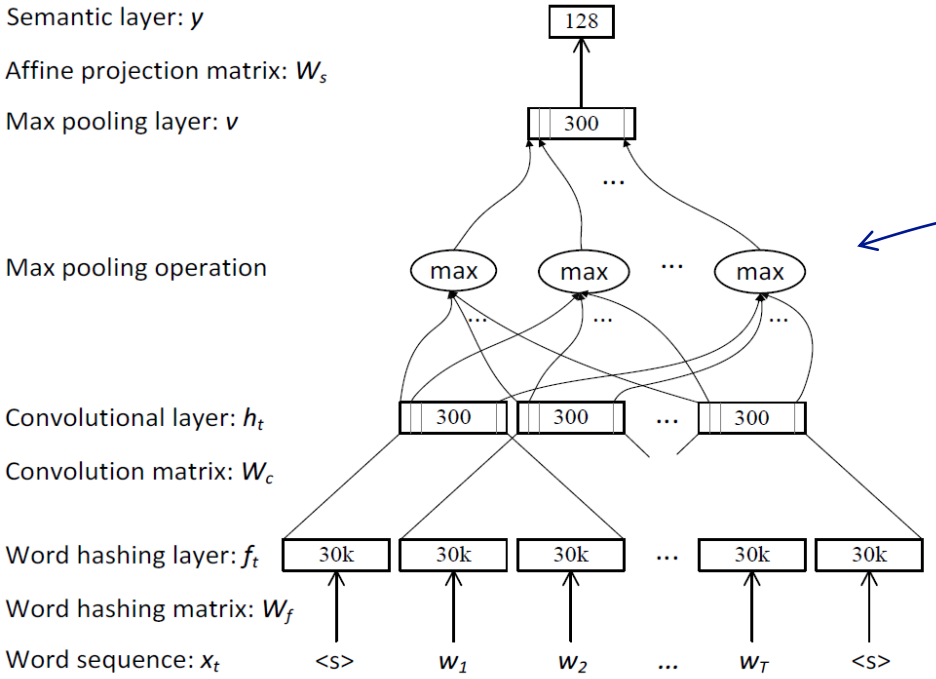


Figure 1: Illustration of the C-DSSM. A convolutional layer with the window size of three is illustrated.

Model local context at the convolutional layer

Model global context at the pooling layer

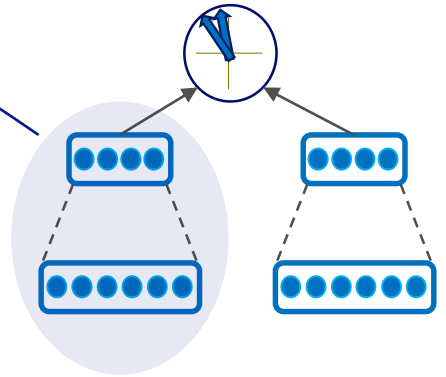


Figure credit [Shen, He, Gao, Deng, Mesnil, WWW2014]

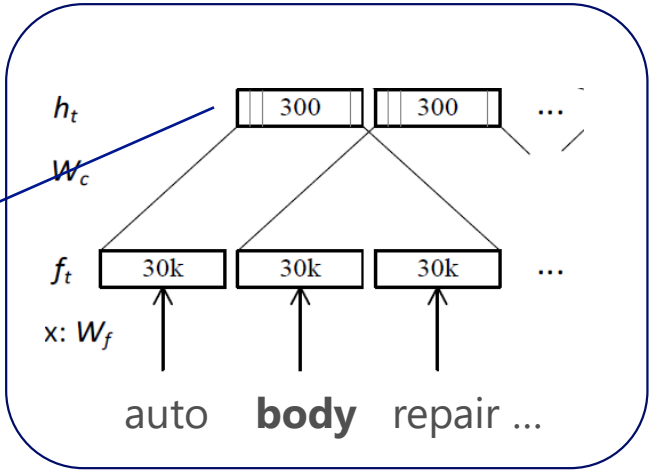
Strong performance on many NLP tasks

Information Retrieval: [Shen, He, Gao, Deng, Mesnil, WWW2014 & CIKM2014], Entity Ranking: [Gao, Pantel, Gamon, He, Deng, Shen, EMNLP2014], Question answering: [Yih, He, Meek, ACL2014; Yih, Chang, He, Gao, ACL2015], Recommendation [Elkahky, Song, He, WWW2015], Spoken language understanding [Chen, Hakkani-Tür, He, ICASSP2016]…

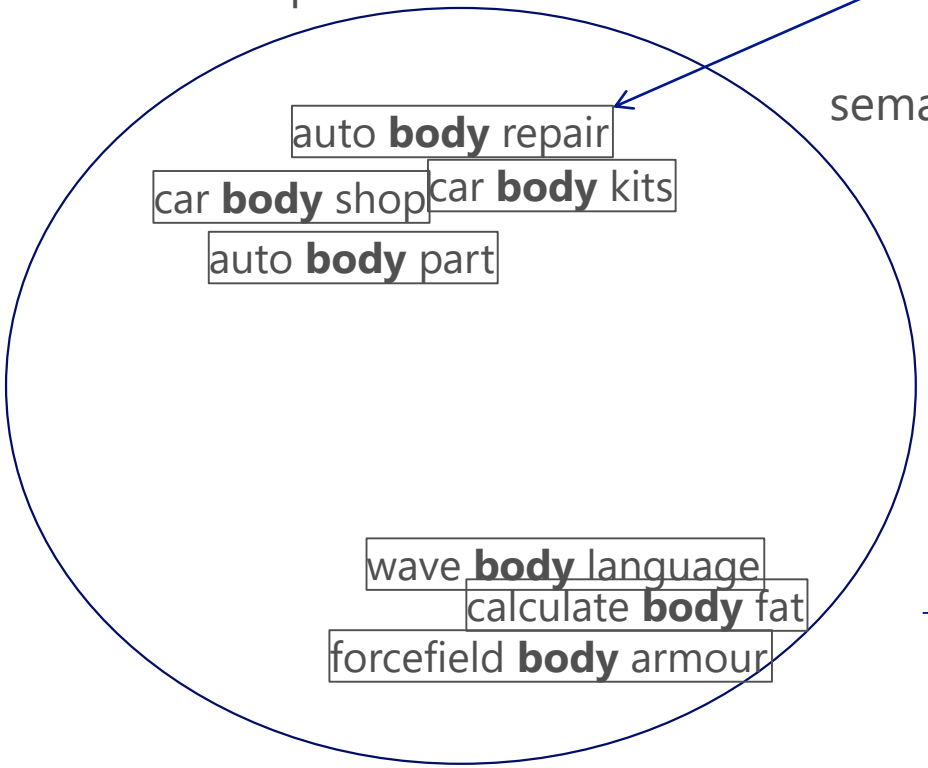
– What does the model learn at the convolutional layer?

Capture the local context dependent word sense

- Learn one embedding vector for each local context-dependent word



$$h_t = W_c \times [f_{t-1}, f_t, f_{t+1}]$$

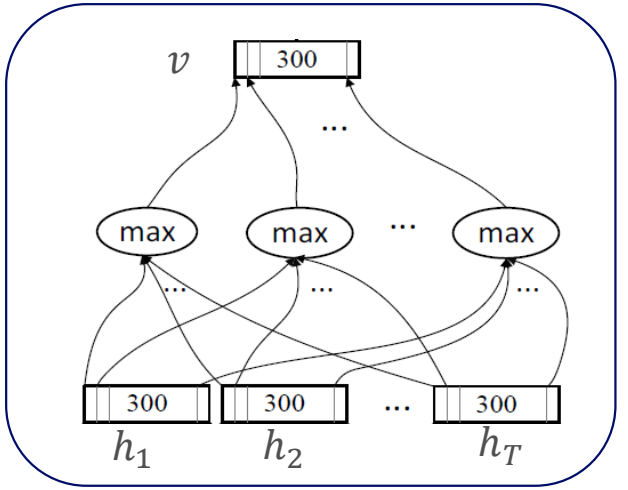


The similarity between different "***body***" within contexts

car **body** shop	**cosine similarity**	
car **body** kits	0.698	**high similarity**
auto **body** repair	0.578	
auto **body** parts	0.555	
wave **body** language	0.301	**low similarity**
calculate **body** fat	0.220	
forcefield **body** armour	0.165	

CDSSM: What happens at the max-pooling layer?

- Aggregate *local topics* to form the *global intent*
- Identify salient words/phrase at the max-pooling layer



$$v(i) = \max_{t=1,\dots,T} \{h_t(i)\}$$

where $i = 1, \dots, 300$

Words that win the most active neurons at the **max-pooling layers:**

auto body repair cost calculator software

Usually, those are salient words containing clear intents/topics

DSSM for Information Retrieval

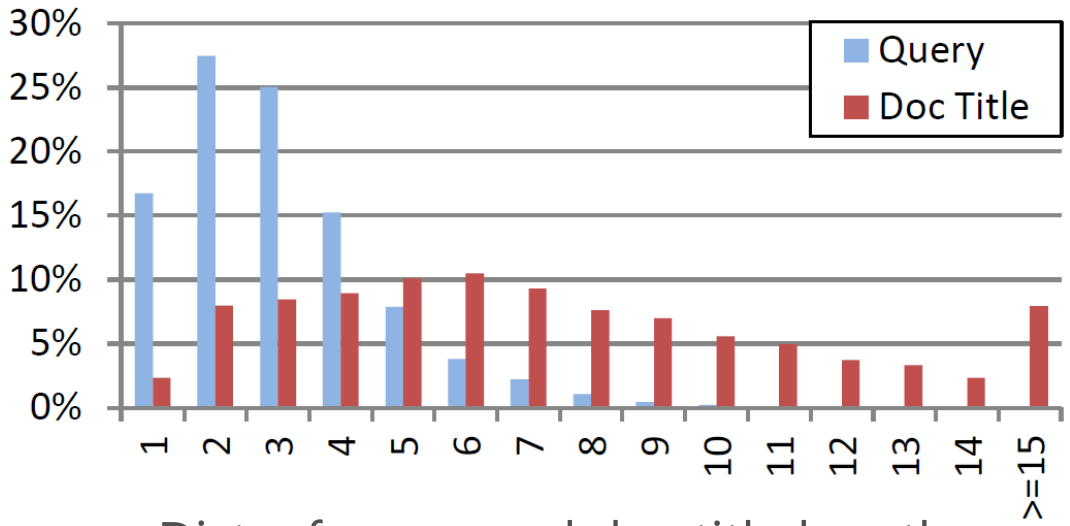
- Training Dataset
  - Mine semantically-similar text pairs from Search Logs, e.g., 30 Million (Query, Document) Click Pairs

QUERY (Q)	Clicked Doc Title (T)
how to deal with stuffy nose	best home remedies for cold and flu
stuffy nose treatment	best home remedies for cold and flu
cold home remedies	best home remedies for cold and flu
… …	… …
go israel	forums goisrael community
skate at wholesale at pr	wholesale skates southeastern skate supply
breastfeeding nursing blister baby	clogged milk ducts babycenter

[Gao, He, Nie, CIKM2010]

Experimental Setting

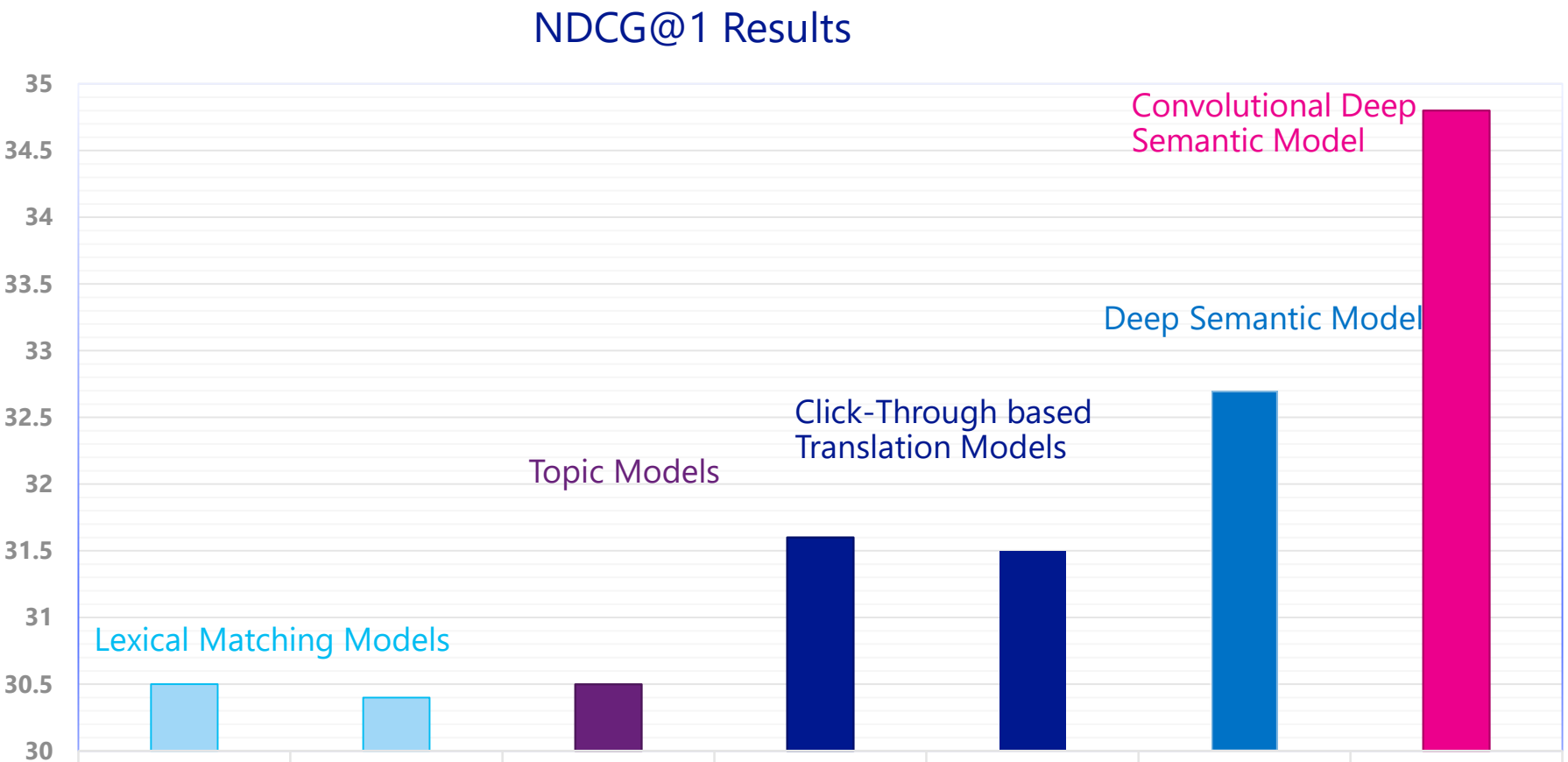
- Testing Dataset
  - **12,071** English queries
  - around 65 web document associated to each query in average
  - Human gives each <query, doc> pair the label, with range **0 to 4**
  - 0: Bad 1: Fair 2: Good 3: Perfect 4: Excellent
- Evaluation Metric: (higher the better)
  - NDCG
- Using NVidia GPU K40 for training



Dist. of query and doc title length

Results

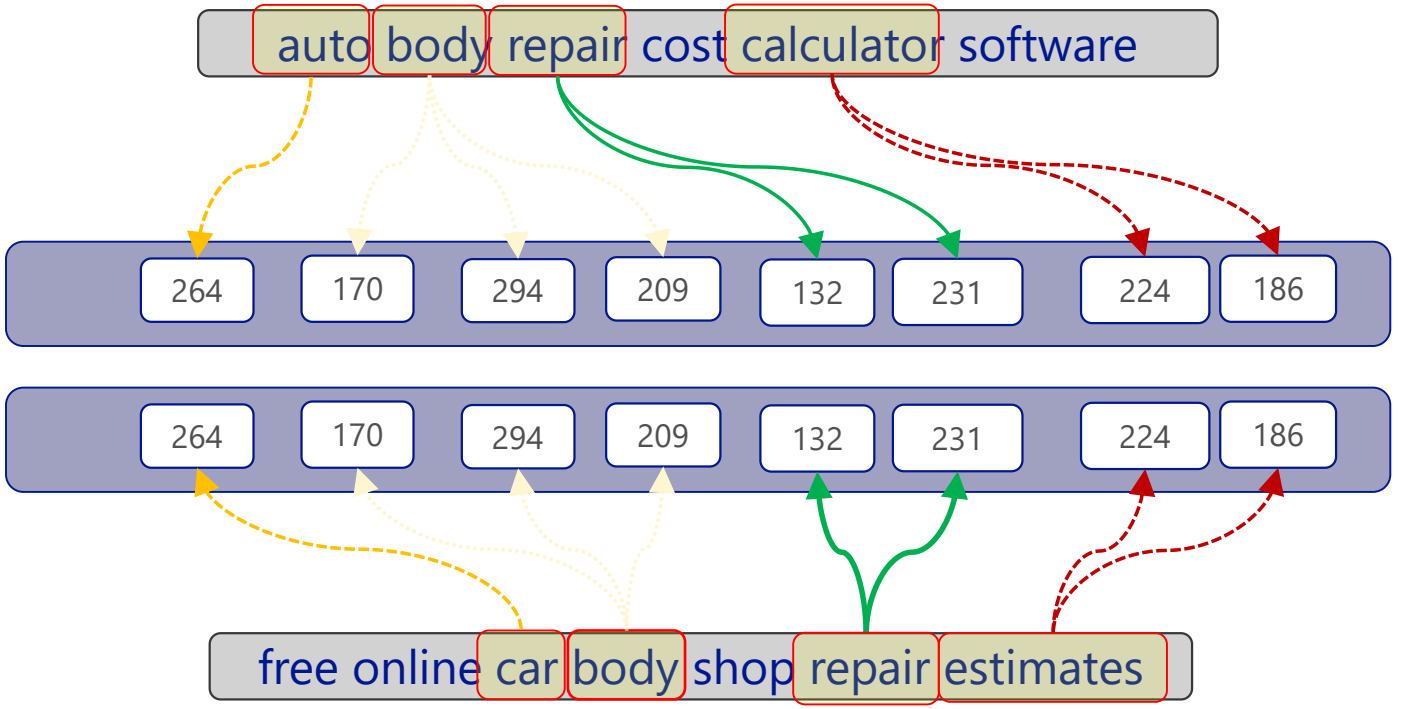
[Shen et al. CIKM2014]



	BM25	ULM	PLSA	BLTM	WTM	DSSM	CDSSM
NDCG@1	30.5	30.4	30.5	31.6	31.5	32.7	34.8

Example: semantic matching

- Semantic matching of query and document

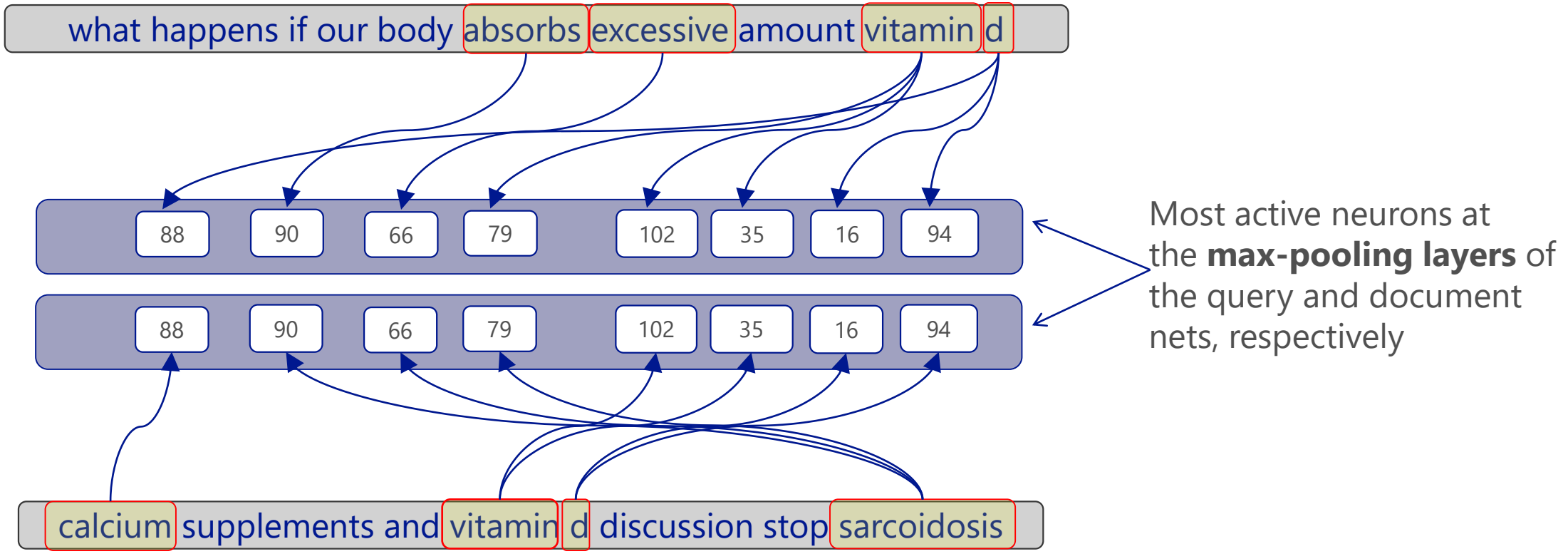


Most active neurons at the **max-pooling layers** of the query and document nets, respectively

More complex semantic matching example

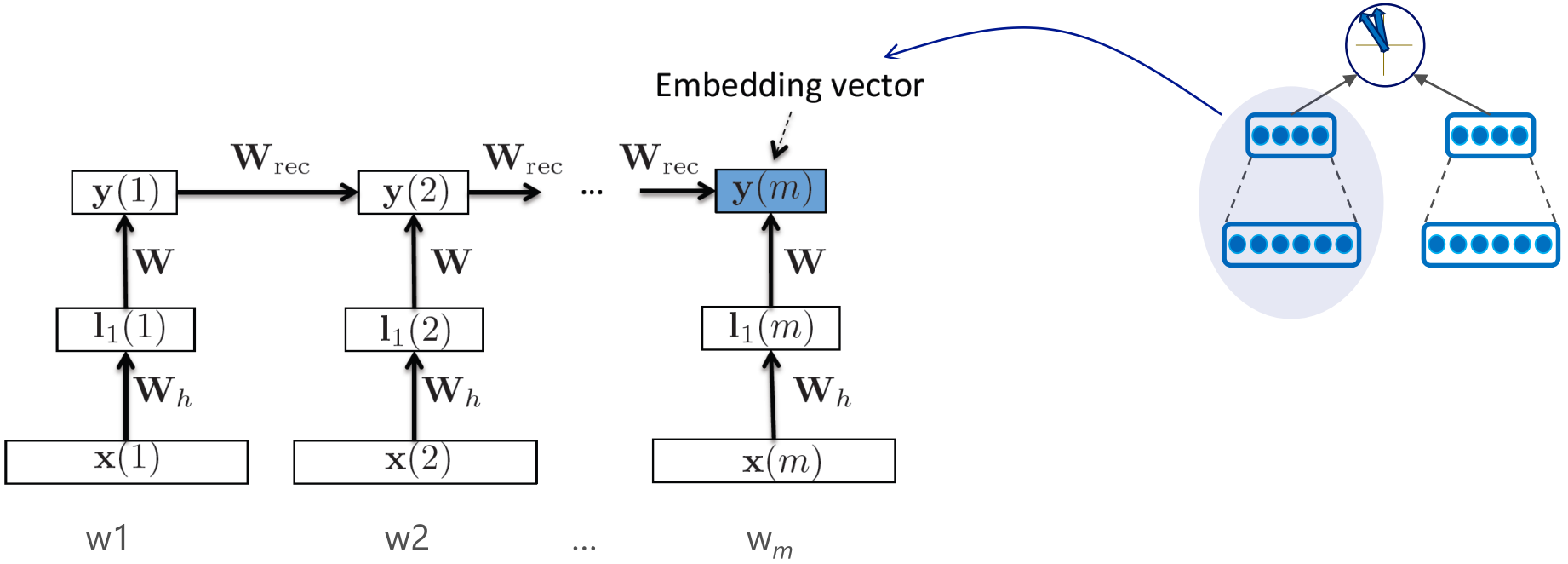
sarcoidosis is a disease, a symptom is excessive amount of calcium in one's urine and blood. So medicines that increase the absorbing of calcium should be avoid. While Vitamin d is closely associated to calcium absorbing.

We observed that "sarcoidosis" in the document title and "absorbs" "excessive" and "vitamin (d)" in the query have high activations at neurons 90, 66, 79, indicating that the model knows that "sarcoidosis" share similar semantic meaning with "absorbs" "excessive" "vitamin (d)", collectively.



Recurrent DSSM

- Encode the word one by one in the recurrent hidden layer
- The hidden layer at the last word codes the semantics of the full sentence
- Model is trained by a cosine similarity driven objective

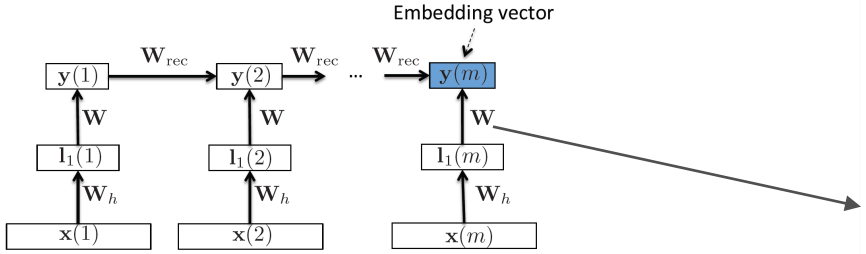


[Palangi, Deng, Shen, Gao, He, Chen, Song, Ward, 2015]

Using LSTM cells

LSTM (long short term memory) uses special cells in RNN

[Hochreiter and J. Schmidhuber, 1997]



$$\mathbf{y}_g(t) = g(\mathbf{W}_4\mathbf{l}_1(t) + \mathbf{W}_{\text{rec4}}\mathbf{y}(t-1) + \mathbf{b}_4)$$
$$\mathbf{i}(t) = \sigma(\mathbf{W}_3\mathbf{l}_1(t) + \mathbf{W}_{\text{rec3}}\mathbf{y}(t-1) + \mathbf{W}_{p3}\mathbf{c}(t-1) + \mathbf{b}_3)$$
$$\mathbf{f}(t) = \sigma(\mathbf{W}_2\mathbf{l}_1(t) + \mathbf{W}_{\text{rec2}}\mathbf{y}(t-1) + \mathbf{W}_{p2}\mathbf{c}(t-1) + \mathbf{b}_2)$$
$$\mathbf{c}(t) = \mathbf{f}(t) \circ \mathbf{c}(t-1) + \mathbf{i}(t) \circ \mathbf{y}_g(t)$$
$$\mathbf{o}(t) = \sigma(\mathbf{W}_1\mathbf{l}_1(t) + \mathbf{W}_{\text{rec1}}\mathbf{y}(t-1) + \mathbf{W}_{p1}\mathbf{c}(t) + \mathbf{b}_1)$$
$$\mathbf{y}(t) = \mathbf{o}(t) \circ h(\mathbf{c}(t)) \qquad (2)$$

where $\circ$ denotes Hadamard (element-wise) product.

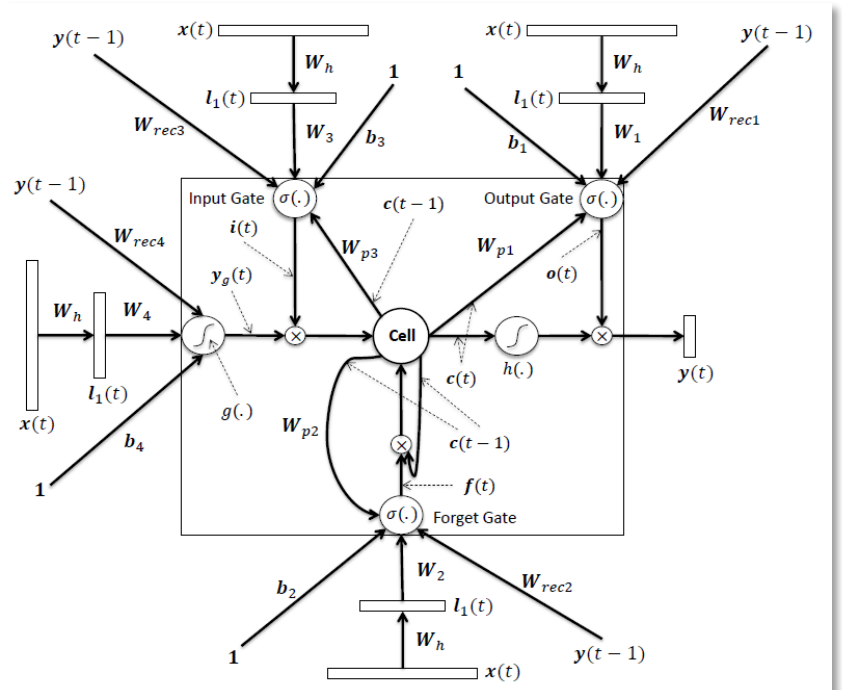


Figure 2. The basic LSTM architecture used for sentence embedding

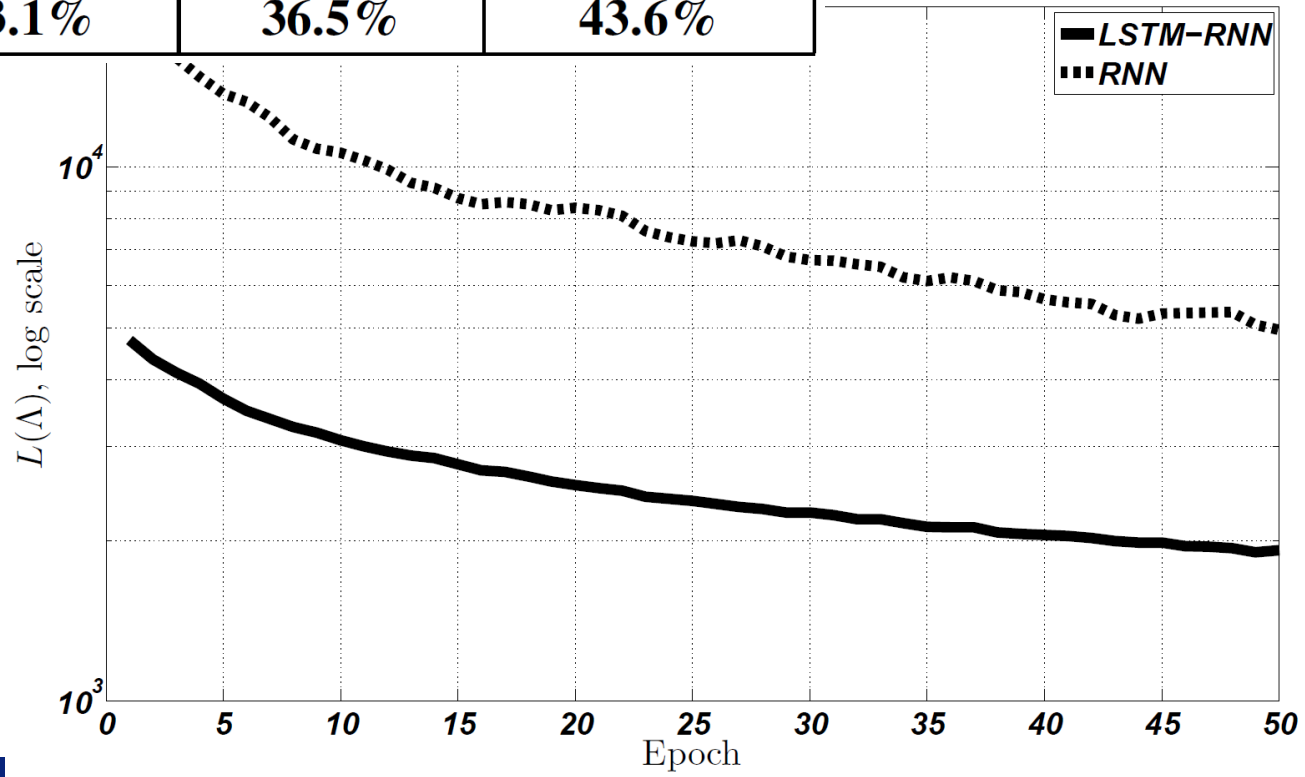
[Palangi, Deng, Shen, Gao, He, Chen, Song, Ward, Deep Sentence Embedding Using the LSTM network: Analysis and Application to IR, IEEE TASL, 2016]

Results

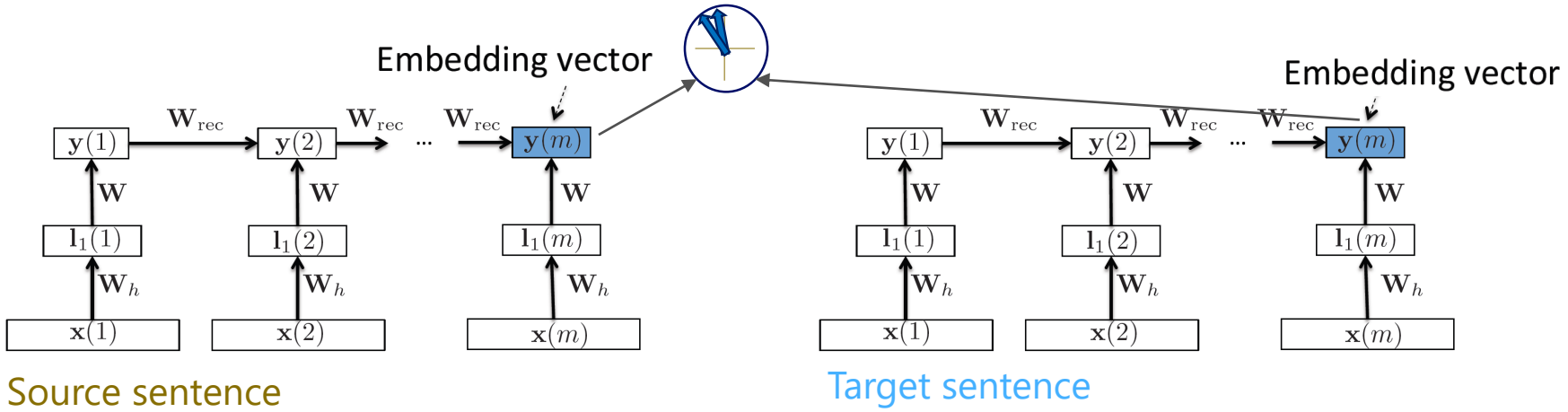
Model	NDCG@1	NDCG@3	NDCG@10
BM25	30.5%	32.8%	38.8%
PLSA (T=500)	30.8%	33.7%	40.2%
DSSM (nhid = 288/96), 2 Layers	31.0%	34.4%	41.7%
CLSM (nhid = 288/96), 2 Layers	31.8%	35.1%	42.6%
RNN (nhid = 288), 1 Layer	31.7%	35.0%	42.3%
LSTM-RNN (ncell = 96), 1 Layer	**33.1%**	**36.5%**	**43.6%**

LSTM learns much faster than regular RNN

LSTM effectively represents the semantic information of a sentence using a vector

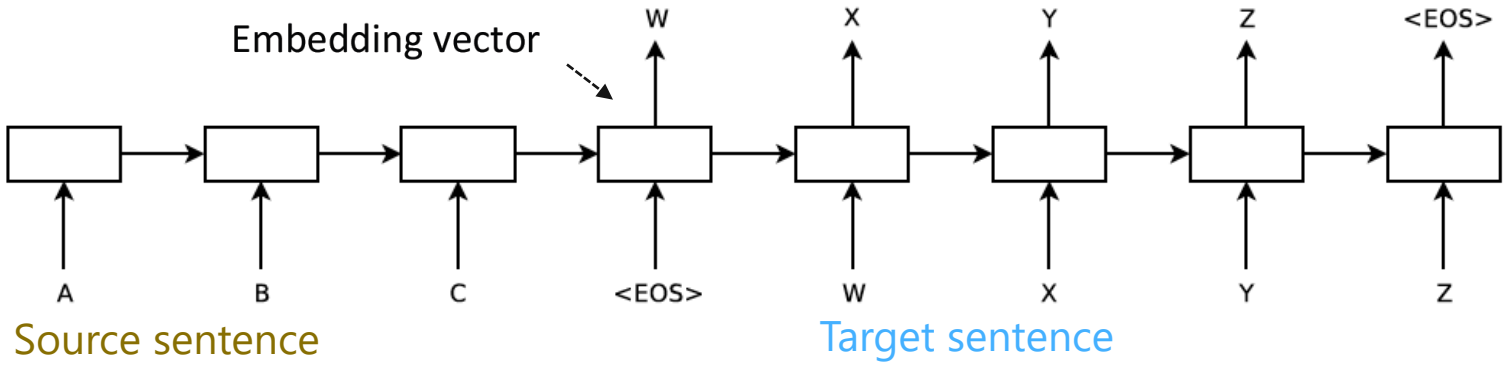


Related work: DSSM vs. Seq2Seq



DSSM optimizes *sentence-level* semantic similarity

vs.

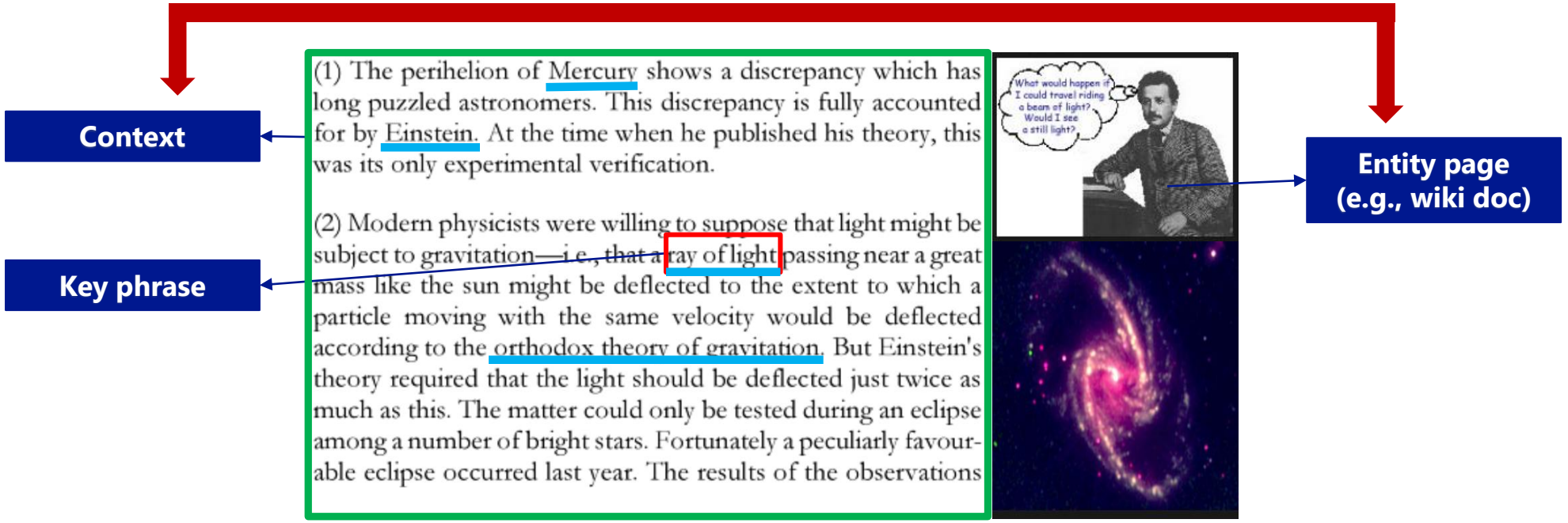


Seq2Seq optimizes *word-level* cross-entropy

[Sutskever, Vinyals, Le, 2014. Sequence to Sequence Learning with Neural Networks]

Contextual Entity Ranking

Given a user-highlighted text span representing an entity of interest, search for supplementary document for the entity



Gao, Pantel, Gamon, He, Deng, Shen, "Modeling interestingness with deep neural networks." EMNLP2014

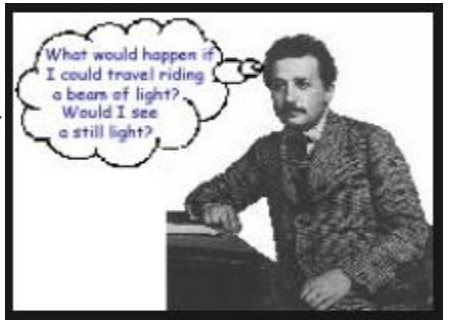
Learning DSSM for contextual entity ranking

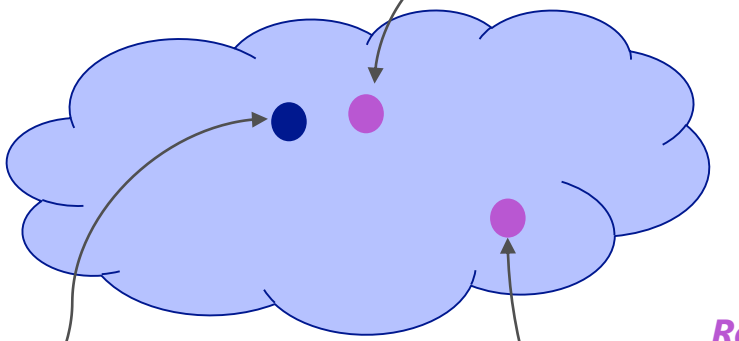
The Einstein Theory of Relativity

(1) The perihelion of Mercury shows a discrepancy which has long puzzled astronomers. This discrepancy is fully accounted for by Einstein. At the time when he published his theory, this was its only experimental verification.

(2) Modern physicists were willing to suppose that light might be subject to gravitation—i.e., that a ray of light passing near a great mass like the sun might be deflected to the extent to which a particle moving with the same velocity would be deflected according to the orthodox theory of gravitation. But Einstein's theory required that the light should be deflected just twice as much as this. The matter could only be tested during an eclipse among a number of bright stars. Fortunately a peculiarly favourable eclipse occurred last year. The results of the observations

Ray of Light (Experiment)





ray of light

Ray of Light (Song)

Extract Labeled Pairs from Web Browsing Logs Contextual Entity Search

- When a hyperlink H points to a Wikipedia P'

http://runningmoron.blogspot.in/

…

I spent a lot of time finding music that was motivating and that I'd also want to listen to through my phone. I could find none. None! I wound up downloading three Metallica songs, a Judas Priest song and one from Bush.

…

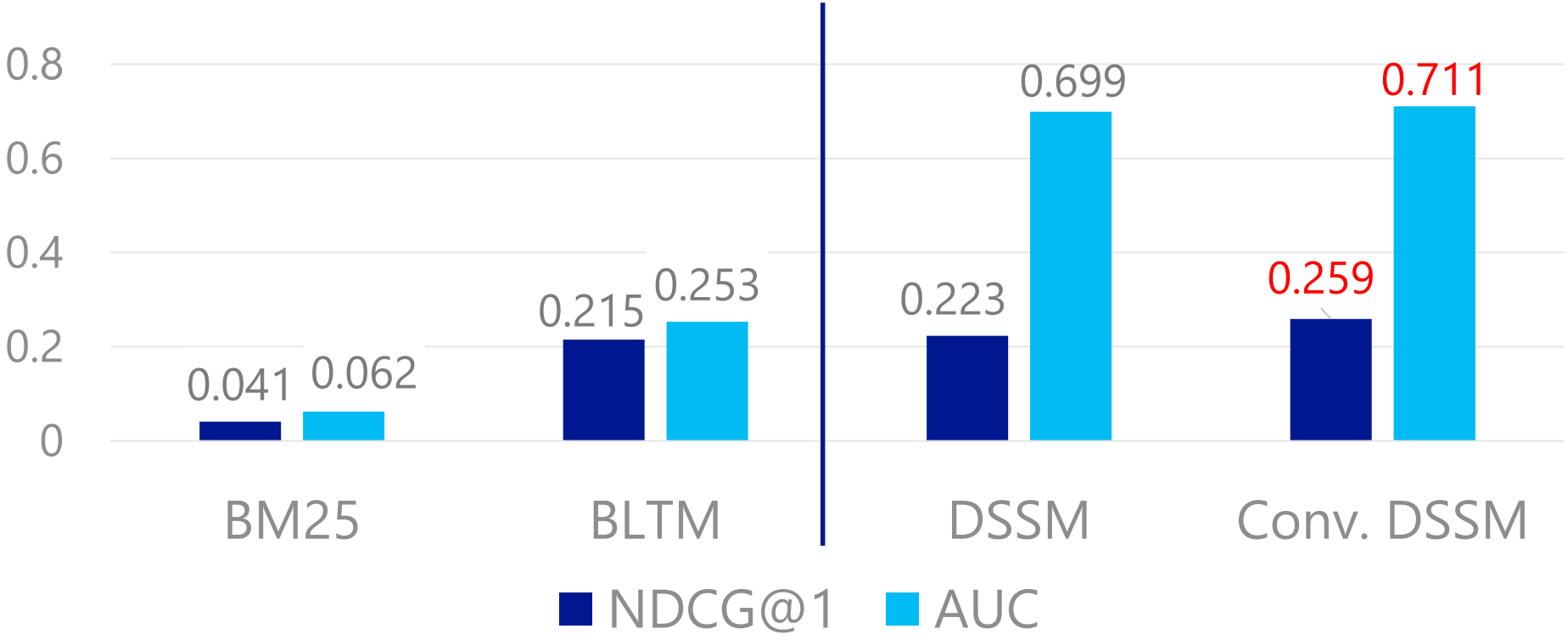
http://en.wikipedia.org/wiki/Bush_(band)

- (anchor text of H & surrounding words, text in P')

Contextual Entity Search: Experimental Settings

- Training/validation data: 18M of user clicks in wiki pages
- Evaluation data
  - Sample 10k Web documents as the source documents
  - Use named entities in the doc as query; retain up to 100 returned documents as target documents
  - Manually label whether each target document is a good page describing the entity
  - 870k labeled pairs in total
- Evaluation metric: NDCG and AUC

Contextual Entity Search Results: DSSM



- DSSM: bag-of-words input
- Conv. DSSM: convolutional DSSM

Some related work

Deep CNN for text input
Mainly classification tasks in the paper

[Kalchbrenner, Grefenstette, Blunsom, A Convolutional Neural Network for Modelling Sentences, ACL2014]

Sequence to sequence learning

[Sutskever, Vinyals, Le, 2014. Sequence to Sequence Learning with Neural Networks]

Paragraph Vector
Learn a vector for a paragraph

Quoc Le, Tomas Mikolov, Distributed Representations of Sentences and Documents, in ICML 2014

Recursive NN (ReNN)
Tree structure, e.g., for parsing

[Socher, Lin, Ng, Manning, "Parsing natural scenes and natural language with recursive neural networks", 2011]

Tensor product representation (TPR)
Tree representation

[Smolensky and Legendre: The Harmonic Mind, From Neural Computation to Optimality-Theoretic Grammar, MIT Press, 2006]

Tree-structured LSTM Network
Tree structure LSTM

[Tai, Socher, Manning. 2015. Improved Semantic Representations From Tree-Structured LSTM Networks.]

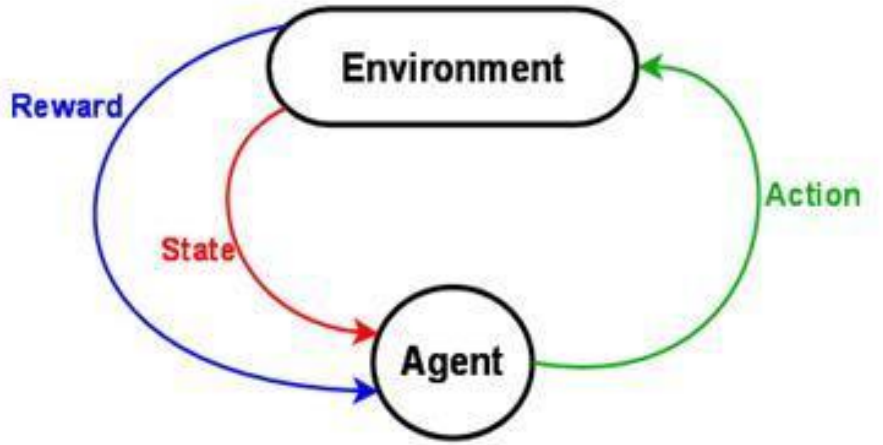
Continuous representations for selected NLP tasks

- Deep semantic similarity model (DSSM) for information retrieval & entity ranking
- Deep reinforcement learning in a continuous semantic space for NLP
- Multimodal semantic learning & inference for image captioning and visual question answering

Background of reinforcement learning

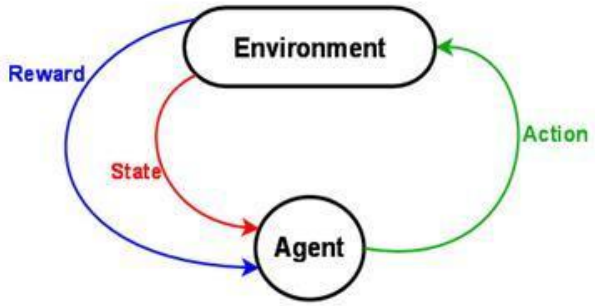
Reinforcement learning model:

- environment state set: *S*
- Action set: *A*
- rules of transitioning between states
- rules that determine the immediate reward of a state transition
- rules that describe what the agent observes



Sutton, Richard S.; Barto, Andrew G. (1998). Reinforcement Learning: An Introduction. MIT Press.

Q-Learning



Used to learn the policy of RL

Policy: a rule that the agent should follow to select actions given the current state

Q-Learning: find optimal policy for Markov decision process (MDP).

Approach: learning an action-value function, a.k.a. Q-function, that computes the expected utility of taking an action in a state – after training converges.

$$Q^{\pi}(s,a) = \mathbb{E}\left\{\sum_{k=0}^{+\infty} \gamma^k r_{t+k} \middle| s_t = s, a_t = a\right\},$$

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \eta_t \cdot \left(r_t + \gamma \cdot \max_a Q(s_{t+1},a) - Q(s_t,a_t)\right)$$

Watkins and Dayan, (1992), 'Q-learning.' Machine Learning.

Recent success

- Deep Q-Network (DQN)

1. Task: playing Atari games
2. RL setting: **huge state space**, e.g., raw image pixels from screen shots. But **small action space**, e.g., possible move of the joystick.
3. Model: using convolutional neural networks to compute $Q(s, a)$.
4. Results: achieve human level performance

action

state

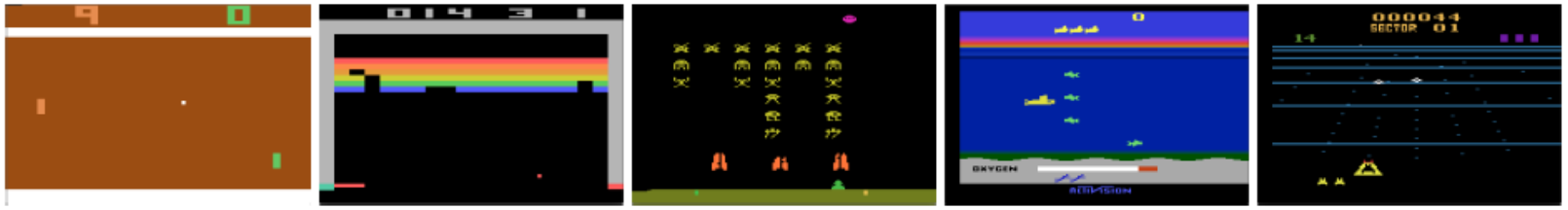


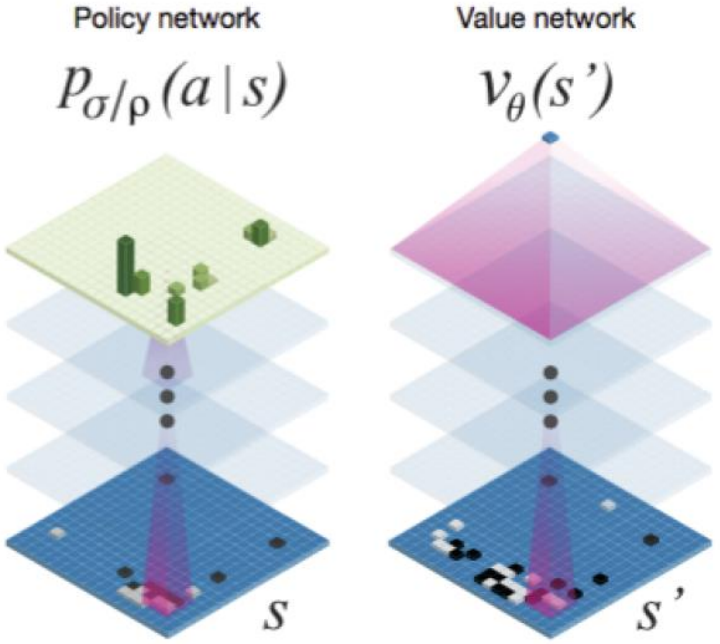
Figure 1: Screen shots from five Atari 2600 Games: (*Left-to-right*) Pong, Breakout, Space Invaders, Seaquest, Beam Rider

Mnih, Kavukcuoglu, Silver, Graves, Antonoglou, Wierstra, Riedmiller, "Playing Atari with Deep Reinforcement Learning", 2013

Recent success (cont.)

- AlphaGo

1. Task: playing Go
2. RL setting: **huge state space**, e.g., 19x19 board (highly complex and sensitive). But still **relatively small action space**, e.g., possible move (one out of <361 positions).
3. Model: built two deep networks: policy network and value network, both are CNNs
4. Use MCTS to estimate the value of states in a search tree.
5. Results: beat world Go Champion



Silver et al, "Mastering the game of Go with deep neural networks and tree search", 2016

Reinforcement learning for language understanding

- Consider the sequential decision making problem for text understanding:
  - E.g., Conversation, Task completion, Playing text-based games…
  - At time t:
    - Agent observes the state as a string of text , e.g., state-text s_t
    - Agent also knows a set of possible actions, each is described as a string text, e.g., action-texts
    - Agent tries to understand the "state text" and all possible "action texts", and takes the **right** action – right means maximizing the long term reward
    - Then, the environment state transits to a new state, agent receives a immediate reward.

[Narasimhan, Kulkarni, Barzilay. 2015]
[He, Chen, He, Gao, Li, Deng, Ostendorf, 2015]

Unbounded action space in RL for NLP

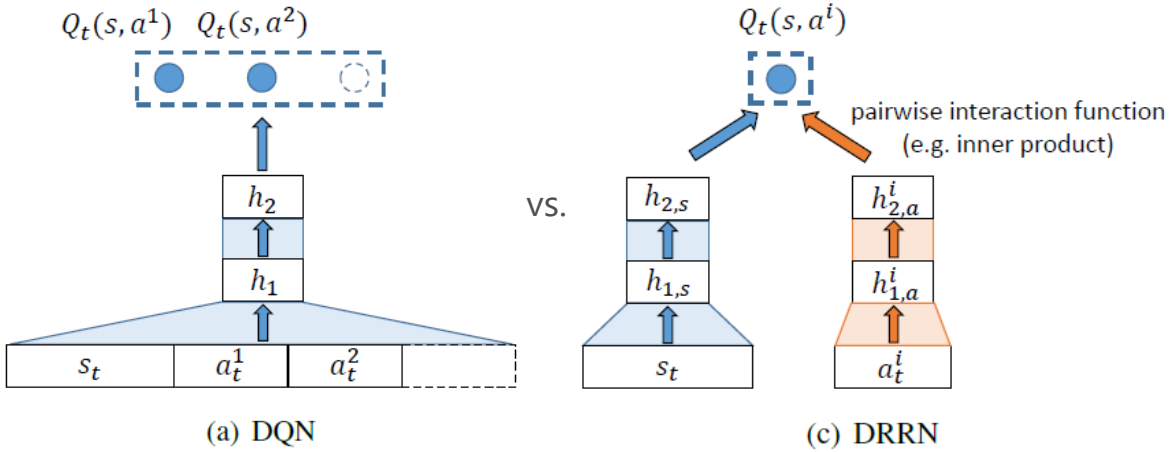
Not only the state space is huge, the action space is huge, too.

– Action is characterized by unbounded natural language description.

Example: a snapshot of a text-based game

Deep Reinforcement Relevance Networks

- Prior DQN work (e.g., Google DeepMind's work on Atari game, AlphaGo): state space unbounded, action space bounded.
- In NLP tasks, usually **the action space**, characterized by natural language, is discrete and nearly **unbounded**.
- We proposed a Deep Reinforcement Relevance Network (**DRRN**)
  - Project both the state and the action into a continuous space
  - Q-function is an relevance function of the state vector and the action vector



Eval metric	Average reward		
hidden dimension	20	50	100
NN-RL (2-hidden)	0.2 (1.2)	2.6 (1.0)	3.6 (0.3)
DQN (2-hidden)	2.5 (1.3)	4.0 (0.9)	5.1 (1.1)
DRRN (2-hidden)	7.3 (0.7)	8.3 (0.7)	**10.5 (0.9)**

Results on text-based game (reward higher the better)

[He, Chen, He, Gao, Li, Deng, Ostendorf, "Deep Reinforcement Learning with a Natural Language Action Space," 2015]

Visualization of the learned continuous space

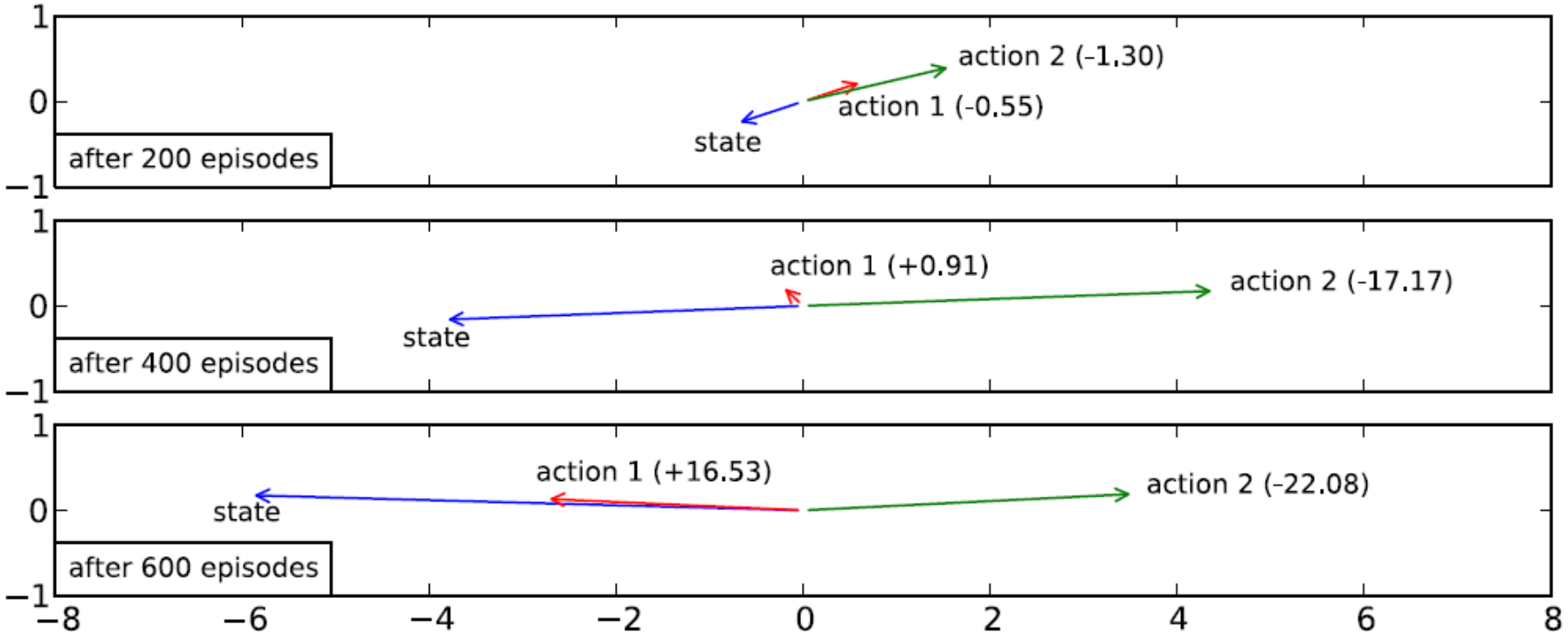
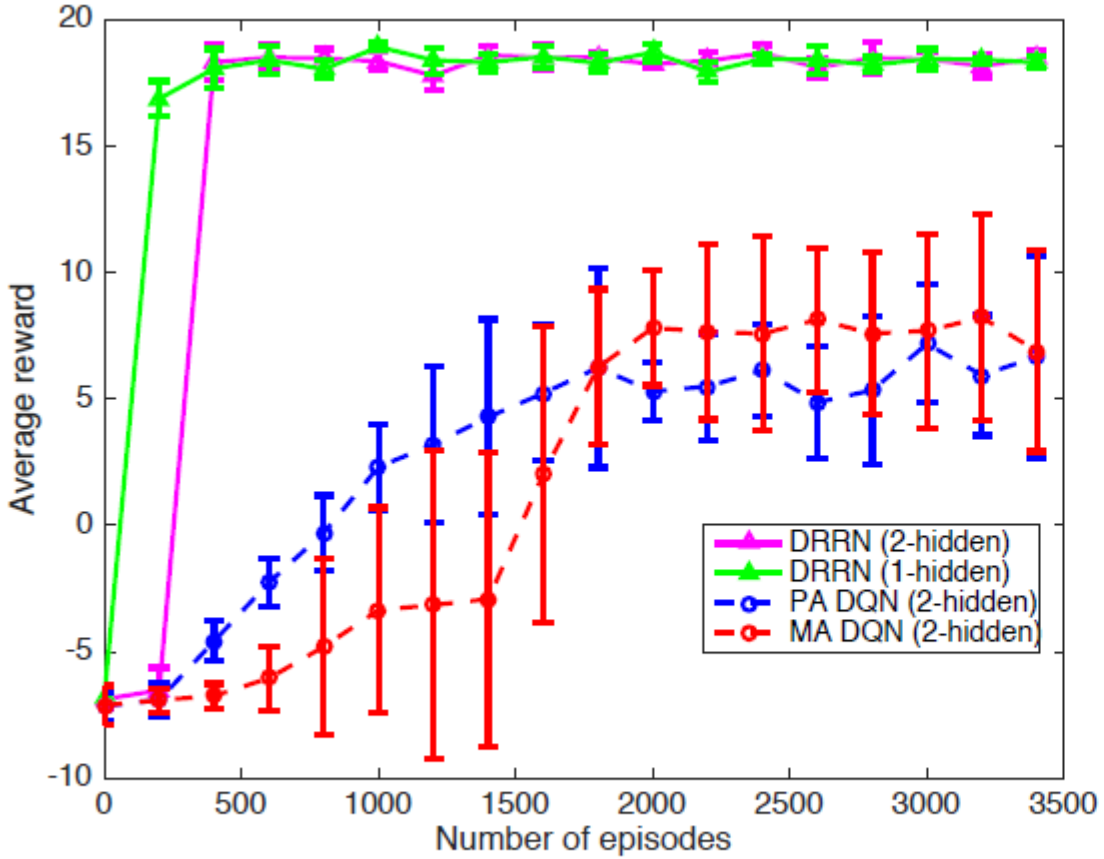
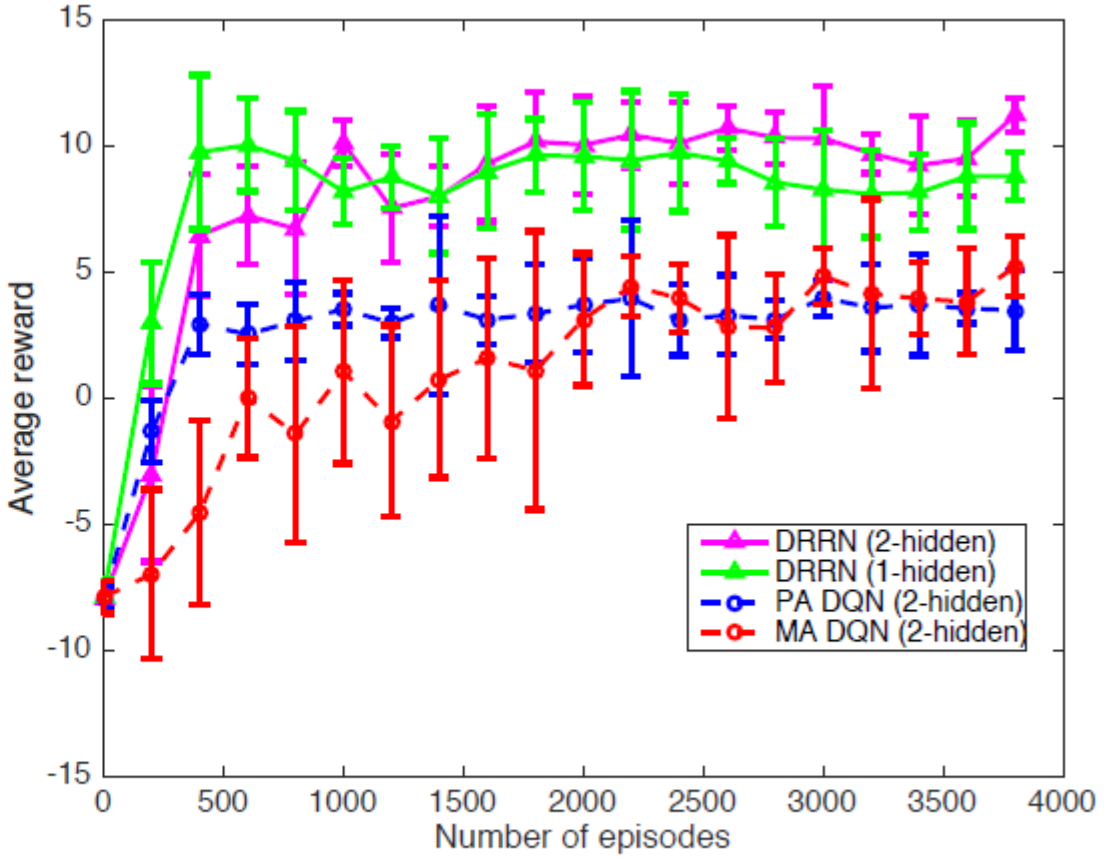


Figure 2: PCA projections of text embedding vectors for state and associated action vectors after 200, 400 and 600 training episodes. The state is "As you move forward, the people surrounding you suddenly look up with terror in their faces, and flee the street." Action 1 (good choice) is "Look up", and action 2 (poor choice) is "Ignore the alarm of others and continue moving forward."

Learning curve: DRRN vs. DQN



(a) Game 1: "Saving John"



(b) Game 2: "Machine of Death"

Tested on two text games

Q-function example values after converged

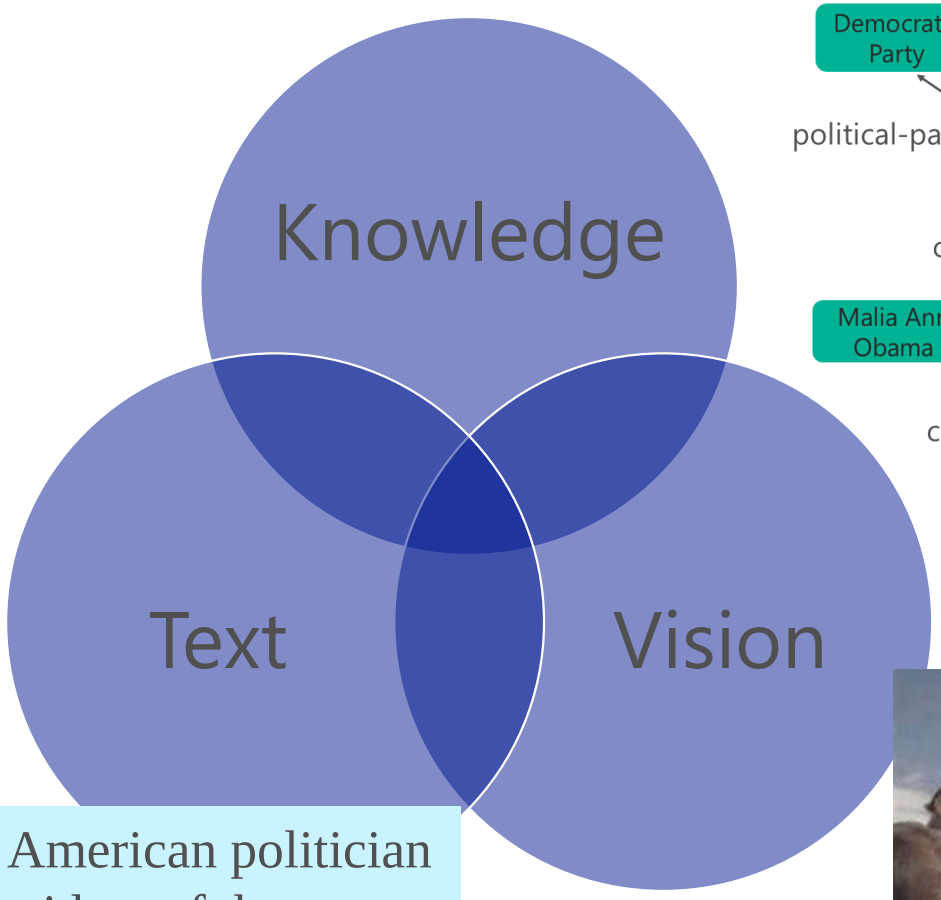
	Text (with predicted Q-values)
State	As you move forward, the people surrounding you suddenly look up with terror in their faces, and flee the street.
Actions in the original game	Ignore the alarm of others and continue moving forward. (-21.5) Look up. (16.6)
Paraphrased actions (not original)	Disregard the caution of others and keep pushing ahead. (-11.9) Turn up and look. (17.5)
Positive actions (not original)	Stay there. (2.8) Stay calmly. (2.0)
Negative actions (not original)	Screw it. I'm going carefully. (-17.4) Yell at everyone. (-13.5)
Irrelevant actions (not original)	Insert a coin. (-1.4) Throw a coin to the ground. (-3.6)

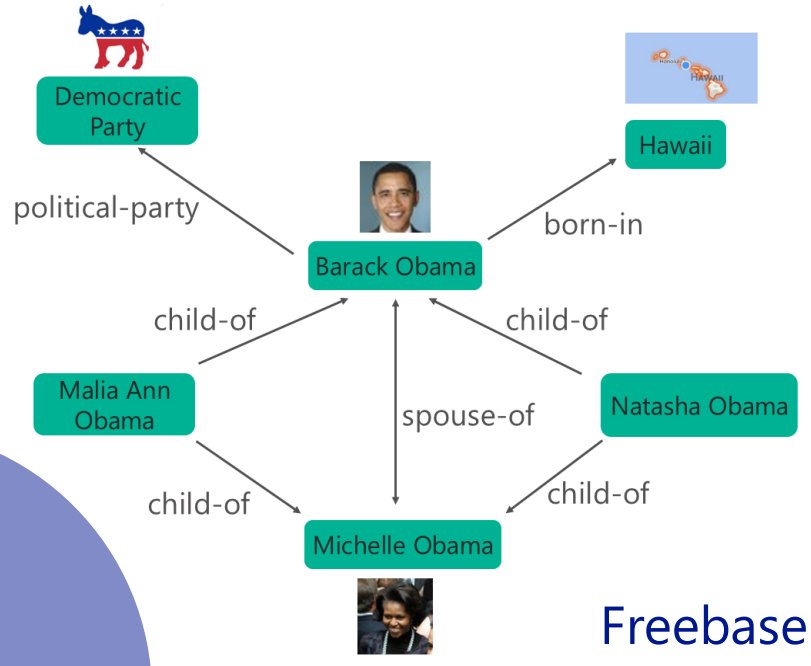
Note that, the DRRN generalizes to unseen actions well, e.g., for these "not original" actions, the model still gives a proper estimate of the Q-value.

Continuous representations for selected NLP tasks

- Deep semantic similarity model (DSSM) for information retrieval & entity ranking
- Deep reinforcement learning in a continuous semantic space for NLP
- Multimodal semantic learning & inference for image captioning and visual question answering

Humans learn to process text, image, and knowledge jointly





Freebase

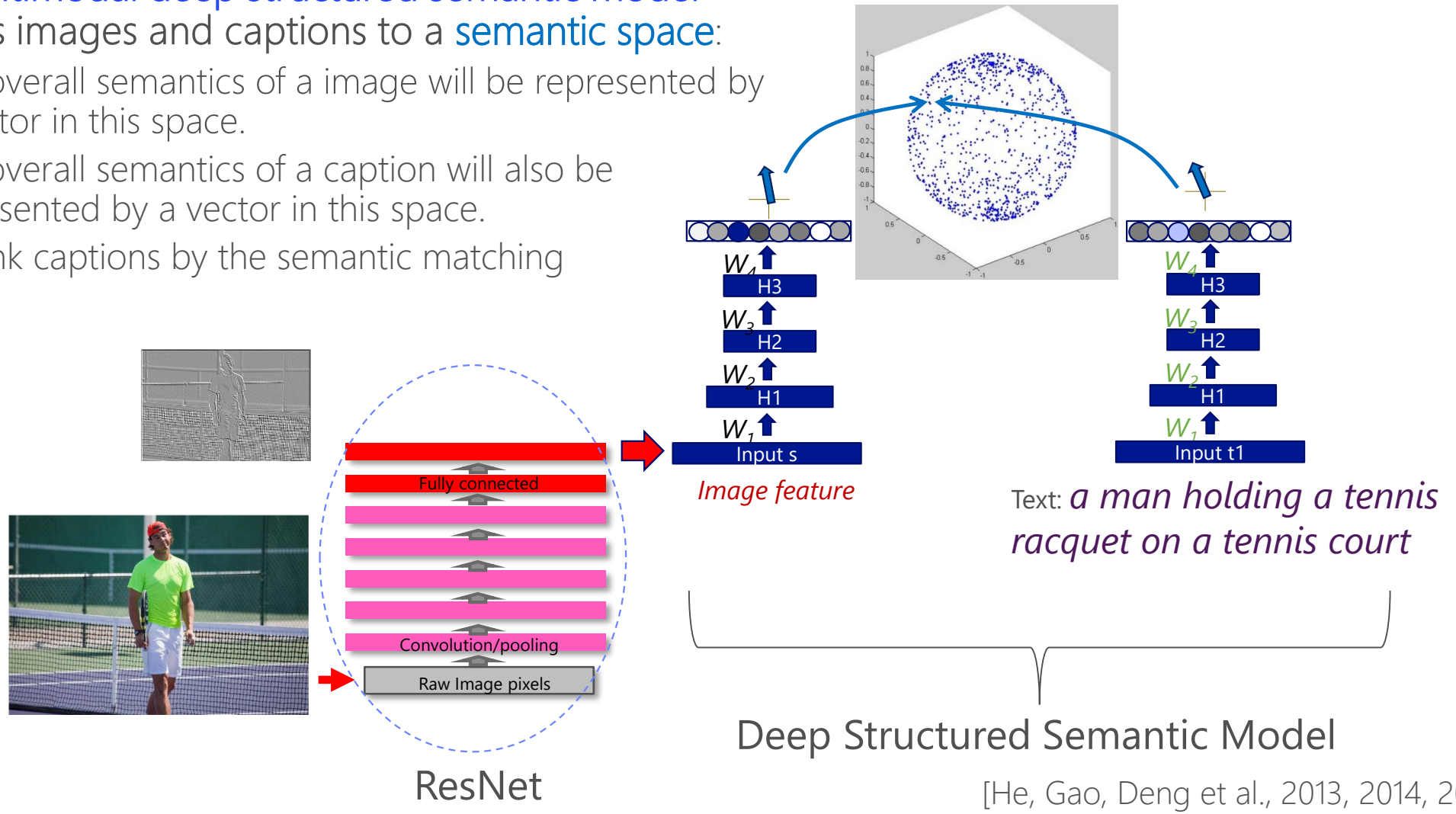
Barack Obama is an American politician serving as the 44th President of the United States. Born in Honolulu, Hawaii, … in 2008, he defeated Republican nominee and was inaugurated as president on January 20, 2009.
(Wikipedia.org)

http://s122.photobucket.com/user/bmeuppls/media/stampede.jpg.html

DSSM: Bridge the gap between image and language!

The multimodal deep structured semantic model projects images and captions to a **semantic space**:

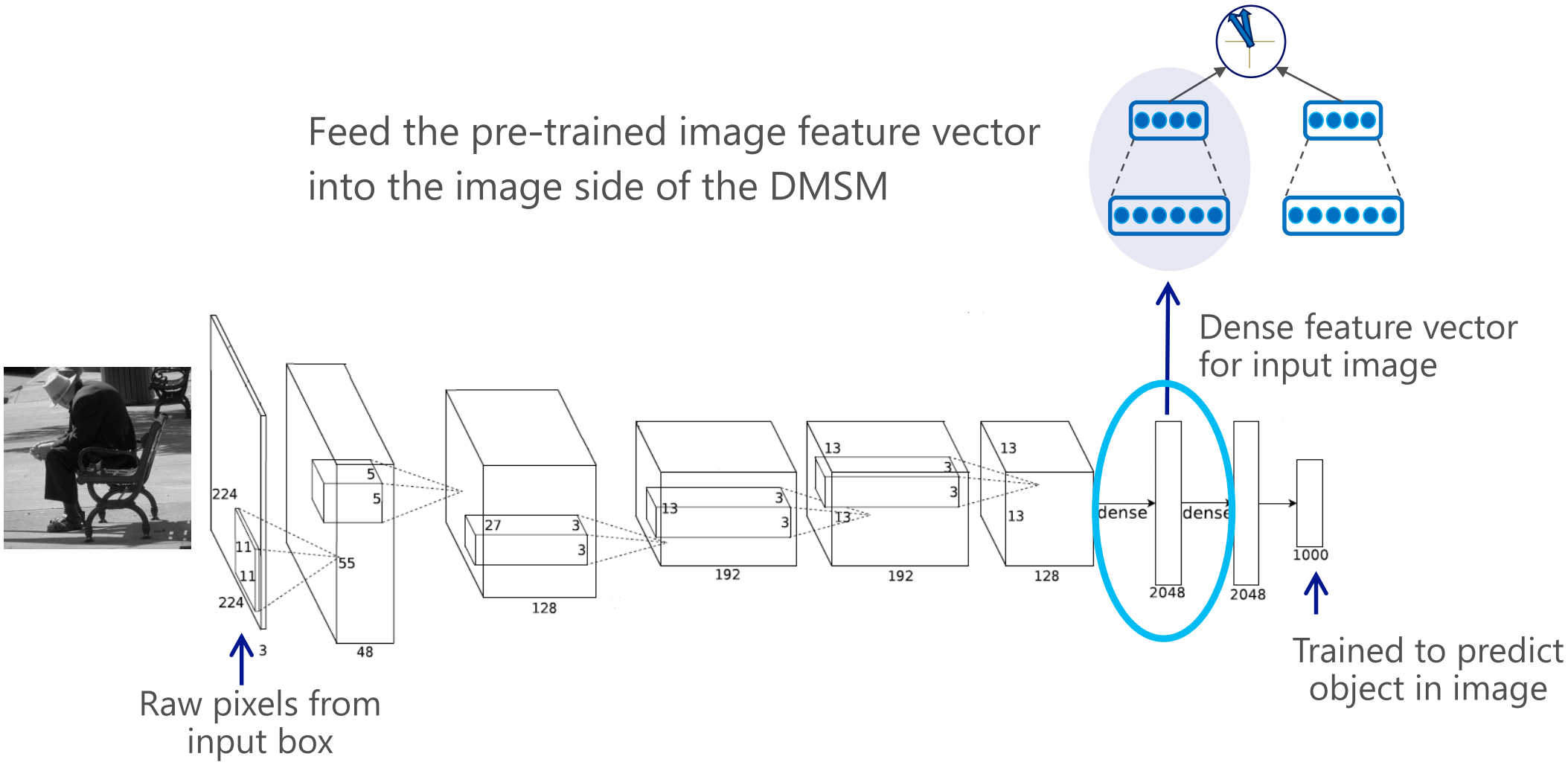
- The overall semantics of a image will be represented by a vector in this space.
- The overall semantics of a caption will also be represented by a vector in this space.
- Rerank captions by the semantic matching



[He, Gao, Deng et al., 2013, 2014, 2015]

The convolutional network at the image side

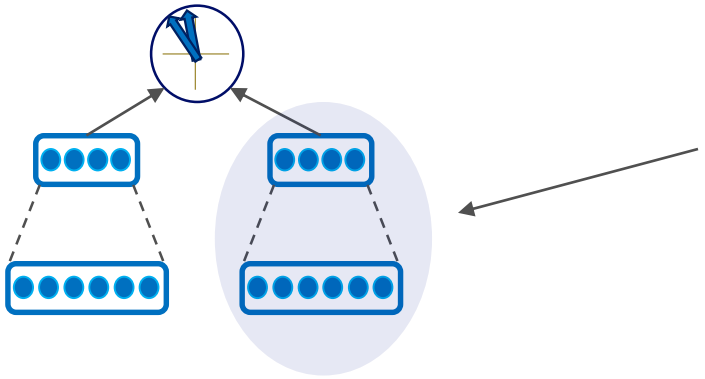
Feed the pre-trained image feature vector into the image side of the DMSM



Tuned image features from AlexNet (Krizhevsky et al., 2012) or VGG (Simonyan and Zisserman, 2014).

The convolutional network at the language side

Models fine-grained structural language information in the caption



Using a convolutional neural network for the text caption side

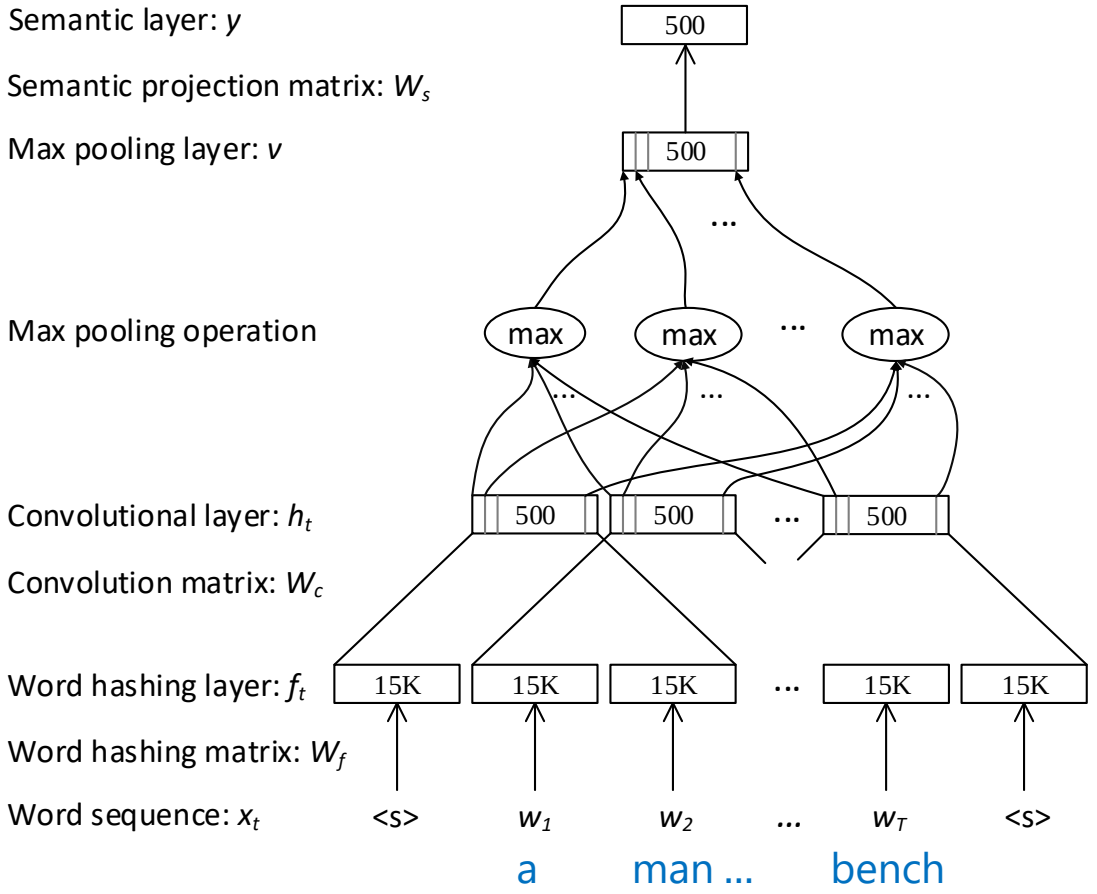


Figure Credit: [Shen, He, Gao, Deng, Mesnil, WWW, April 2014]

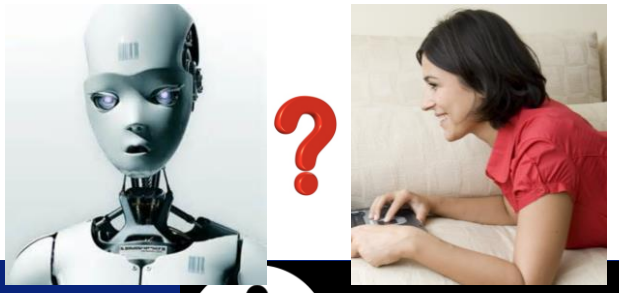
Image Captioning (one step from perception to cognition)

describe objects, attributes, and relationship in an image, in a natural language form

a man holding a tennis racquet on a tennis court

the man is on the tennis court playing a game

-- Let's do a Turing Test!



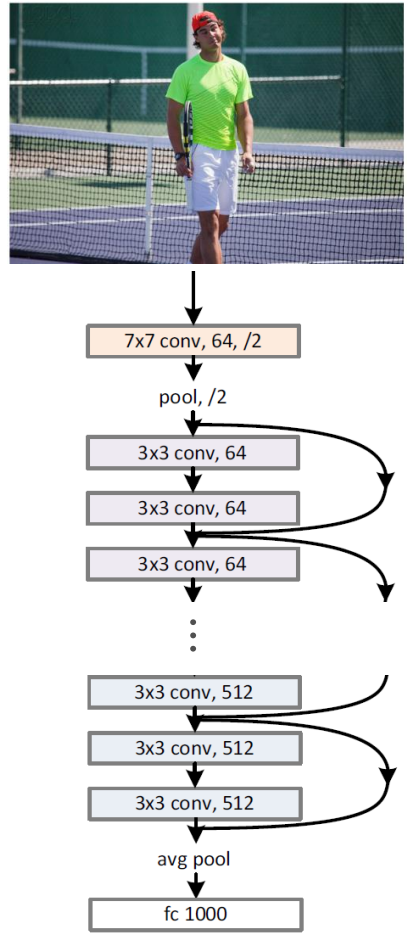
Deep ResNet for visual concepts detection

[He, Zhang, Ren, Sun, CVPR2015]

ResNet

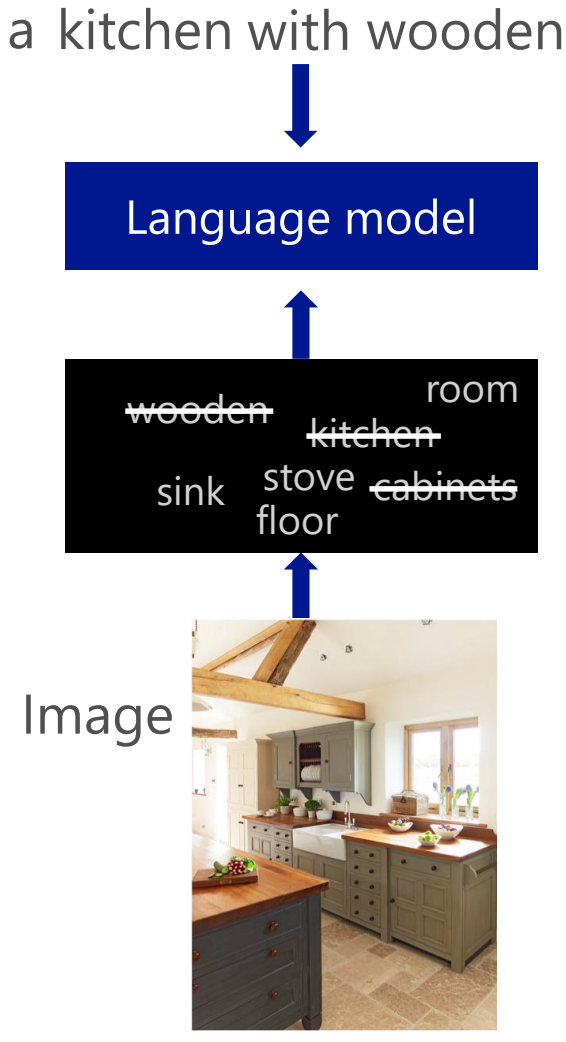
- ImageNet winning solution
- Treat as multiclass problem
- Sigmoid output
- No softmax normalization

Trained on multiple GPUs



man, tennis, court, holding, shirt, yellow, racquet, …

MELM for candidate generation



cabinets

MaxEnt LM

$p(cabinets|with\ wooden)$

a kitchen with wooden cabinets

Beam search to generate 500 candidates

1. wooden cabinets in a kitchen
2. a sink and cabinets

...

500. a room with stove on the floor

[Fang, et al., CVPR 2015]

Image captioning

- Image word detection
  - Deep-learned model to detect key concepts in the image
- Language model generates caption candidates
  - Maxent language model (MELM) conditional on words detected from the image
- Deep multi-modal semantic model re-ranking
  - Hypothetical captions re-ranked by deep-learned multimodal similarity model (DMSM) looking at the entire image

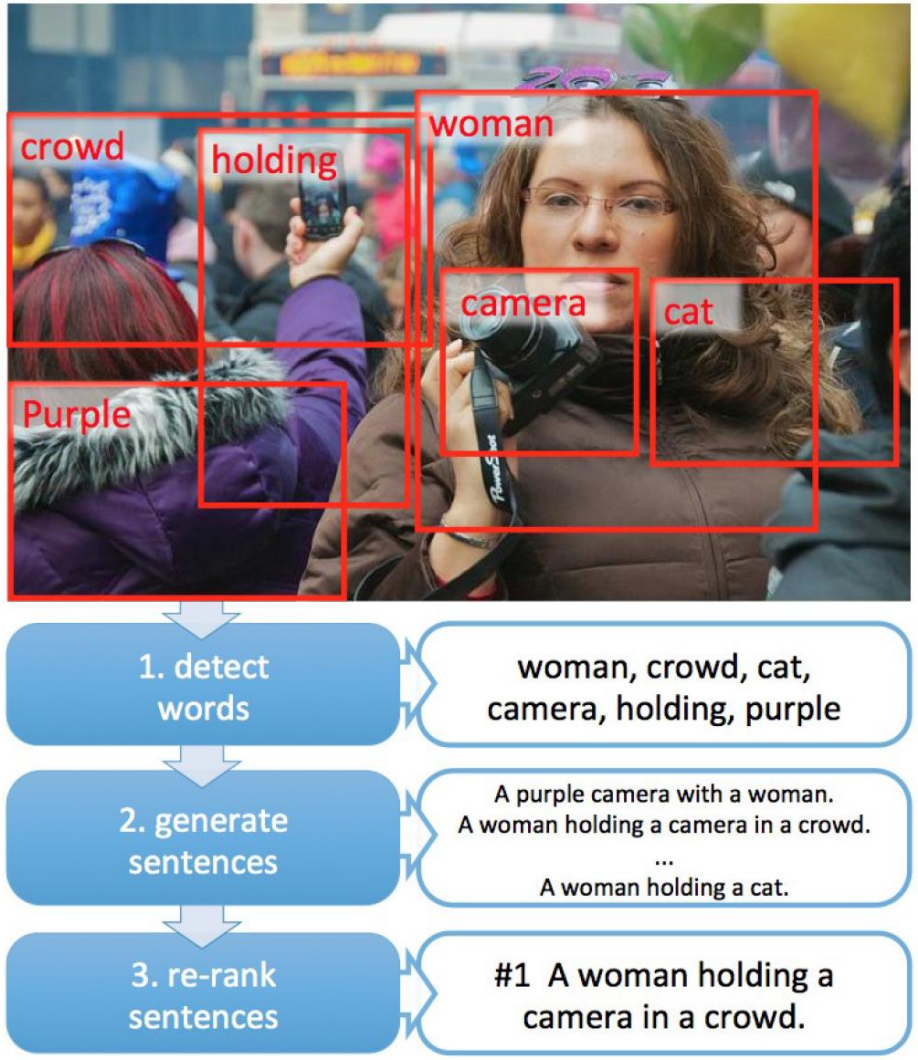


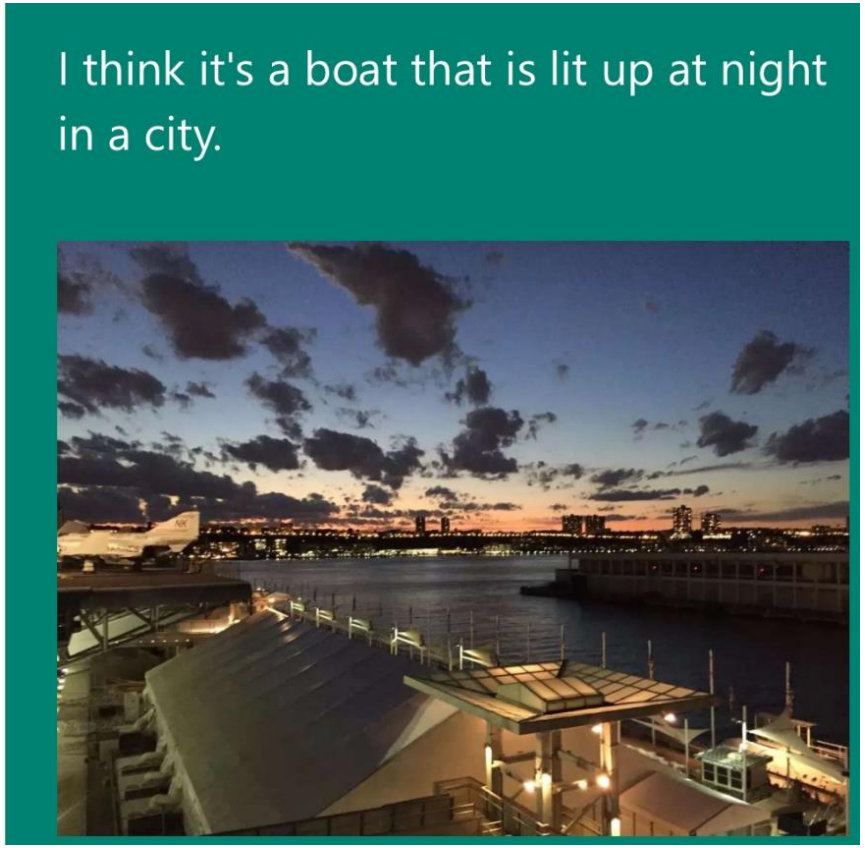
Figure 1. An illustrative example of our pipeline.

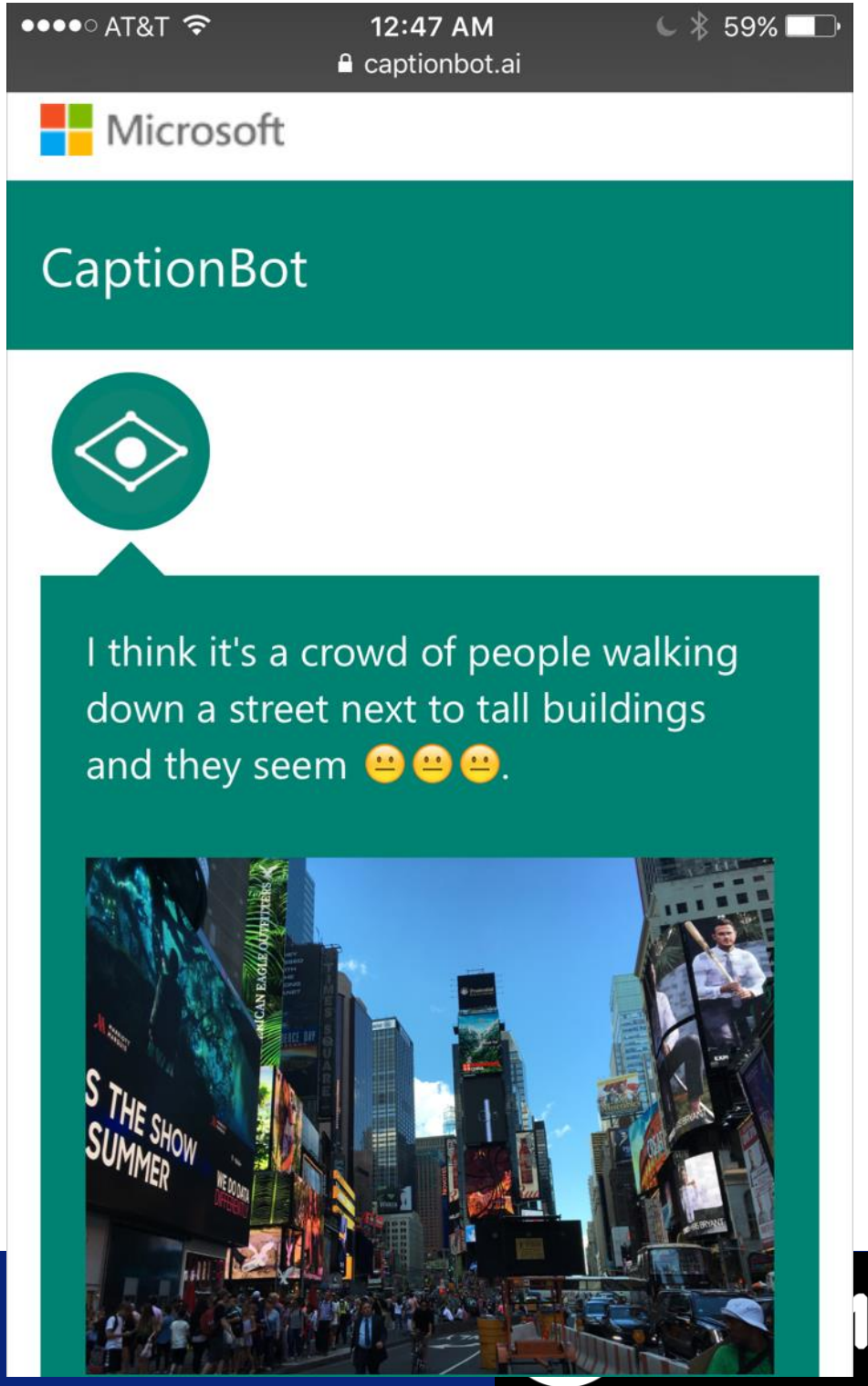
[Fang, Gupta, Iandola, Srivastava, Deng, Dollar, Gao, He, Mitchell, Platt, Zitnick, Zweig, "From Captions to Visual Concepts and Back," CVPR, June 2015]

Public App: CaptionBot

http://CaptionBot.ai

works with any phone/browser





[Tran, He, Zhang, Sun, Carapcea, Thrasher, Buehler, Sienkiewicz, "**Rich Image Captioning in the Wild,**" Deep Vision, CVPR, 2016]

More examples from CaptionBot

CaptionBot

I am not really confident, but I think it's Leonardo da Vinci sitting in front of a mirror and she seems 😐.

y street filled with

I think it's a man standing in front of a building and he seems 😁.

From Captioning to Question Answering

- Answer natural language questions according to the content of a reference image.

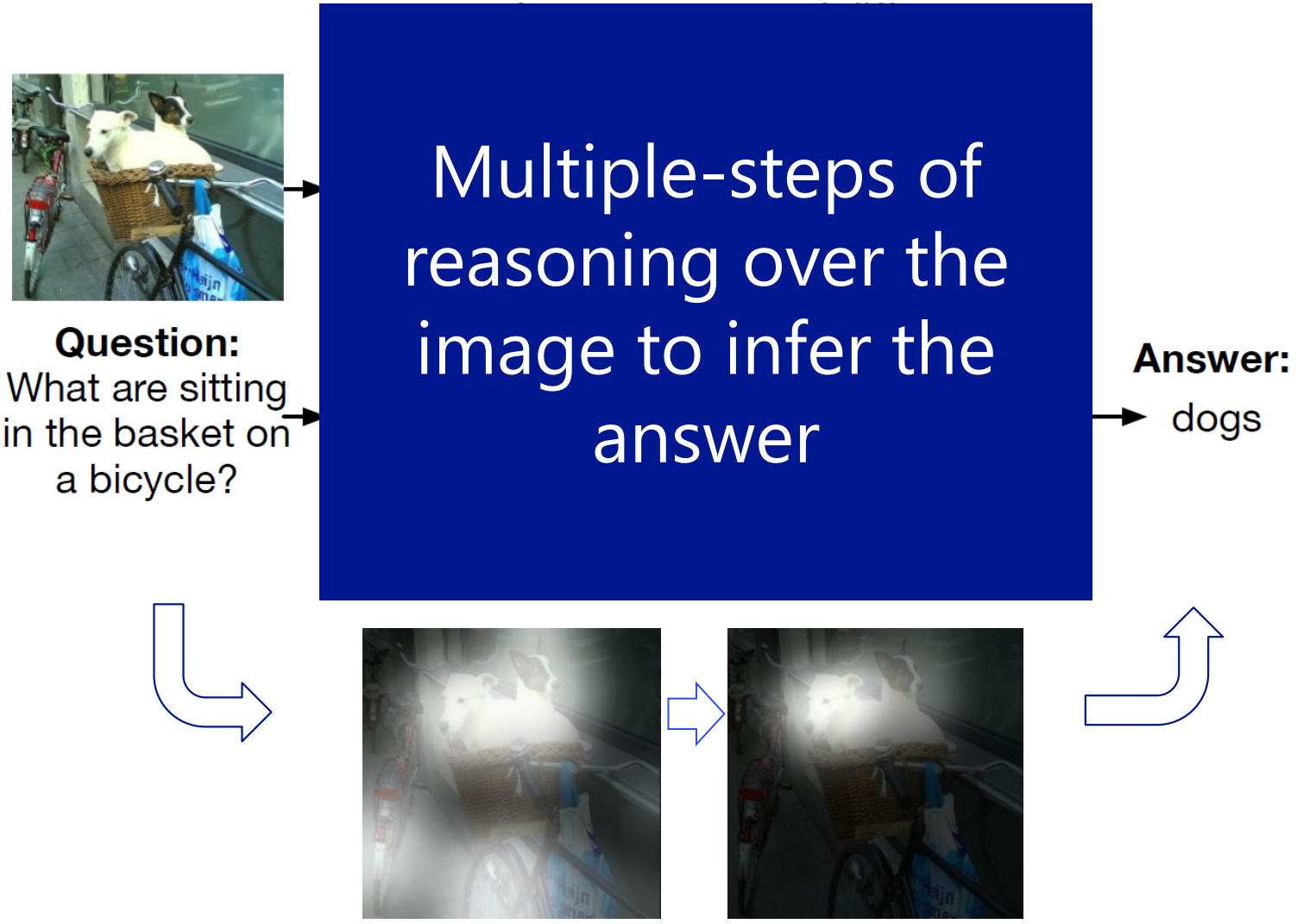
Question:
What are sitting in the basket on a bicycle?

Image Question Answering (IQA)

Answer:
dogs

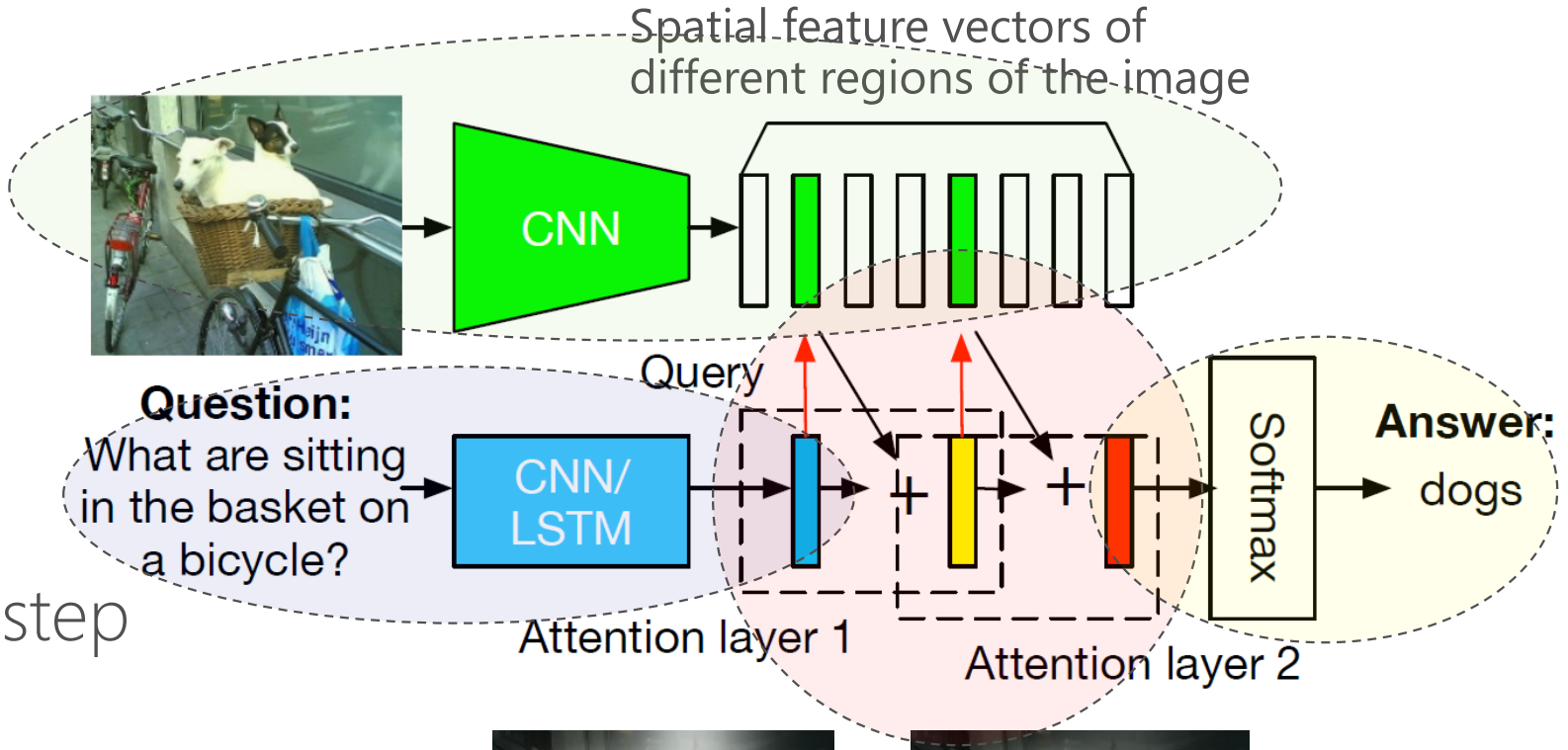
Caption vs. QA: need reasoning

Image QA: reasoning is the key.



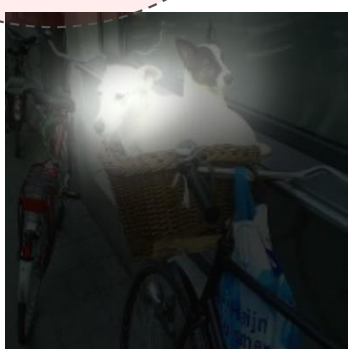
Stacked Attention Networks

[Yang, He, Gao, Deng, Smola, CVPR16]



SANs perform multi-step reasoning

1. Question model
2. Image model
3. Multi-level attention model
4. Answer predictor



1. The image model in the SAN

- Image Model

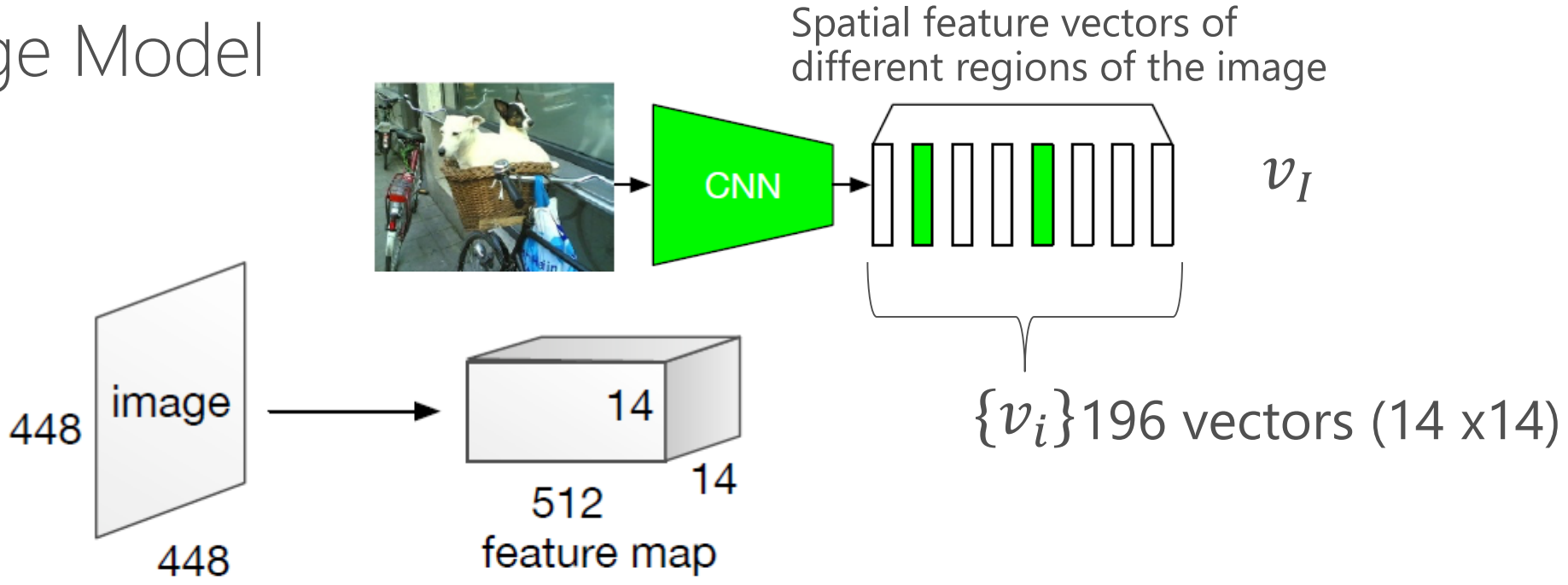


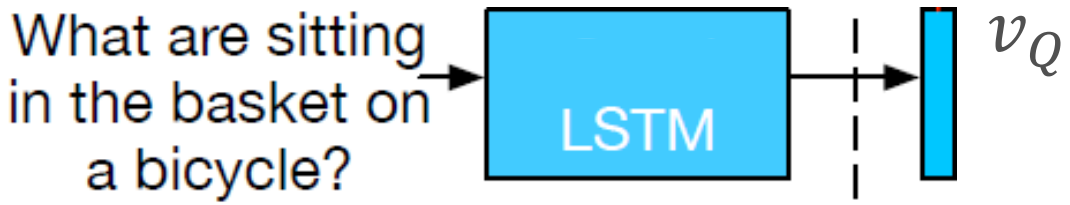
Figure 2: CNN based image model

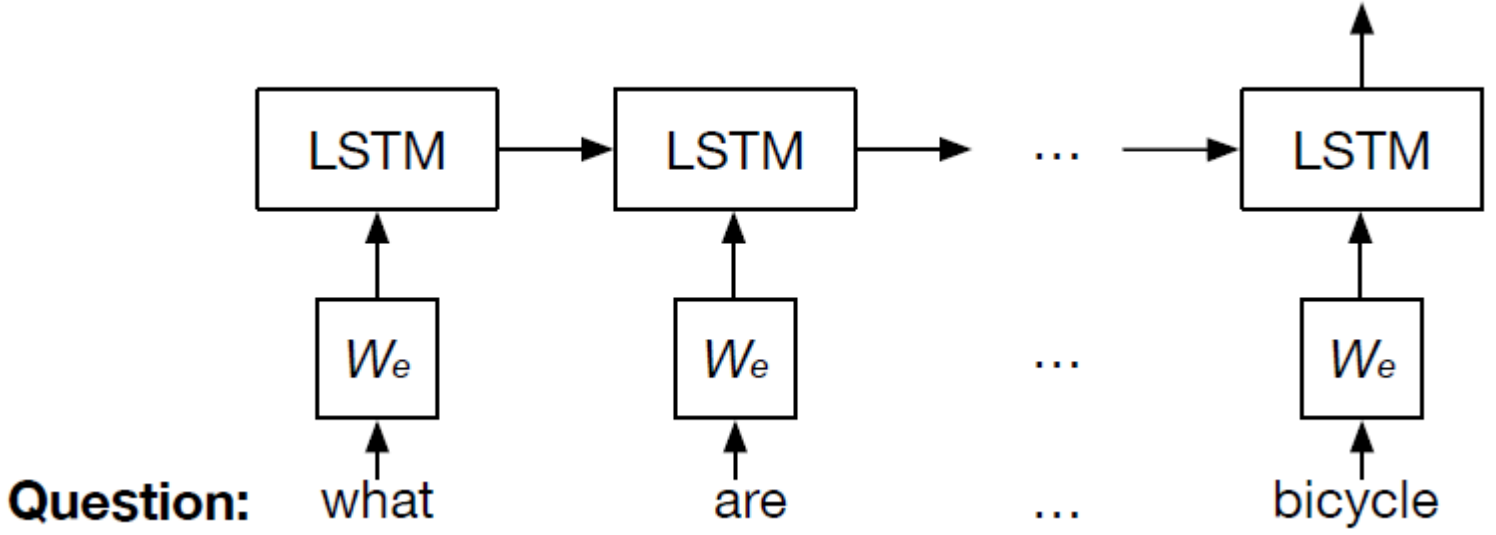
$$f_I = \text{CNN}_{vgg}(I). \qquad v_I = \tanh(W_I f_I + b_I)$$

2. The question model in the SAN

- Question Model

  Code the question into a vector using a LSTM

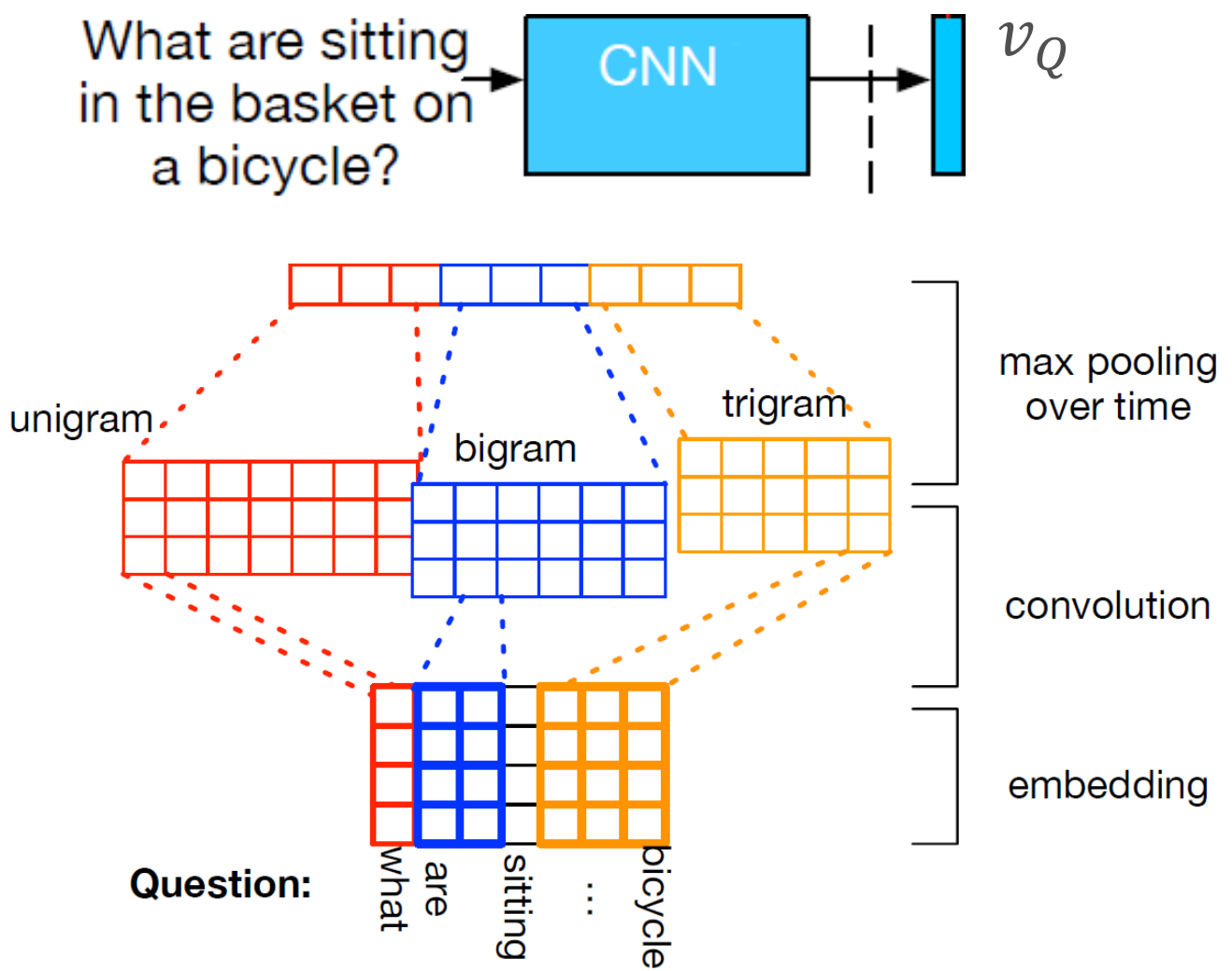




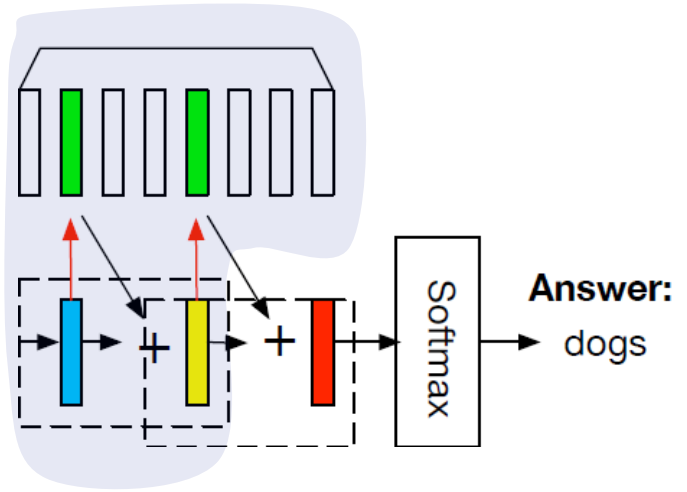
2. The question model in the SAN (alternative)

- Question Model

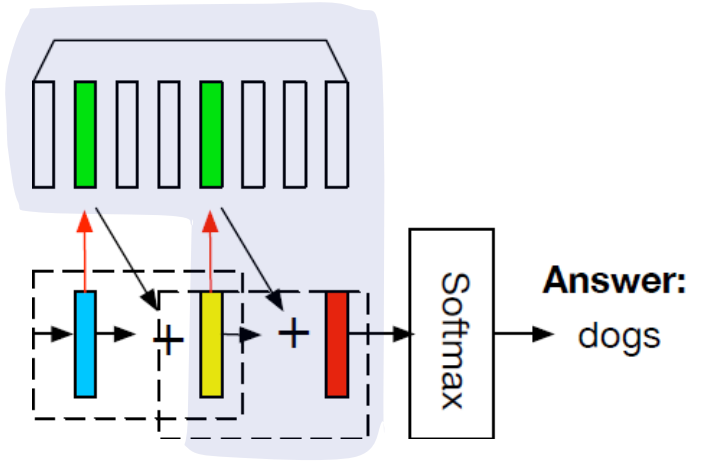
  Code the question into a vector using a CNN

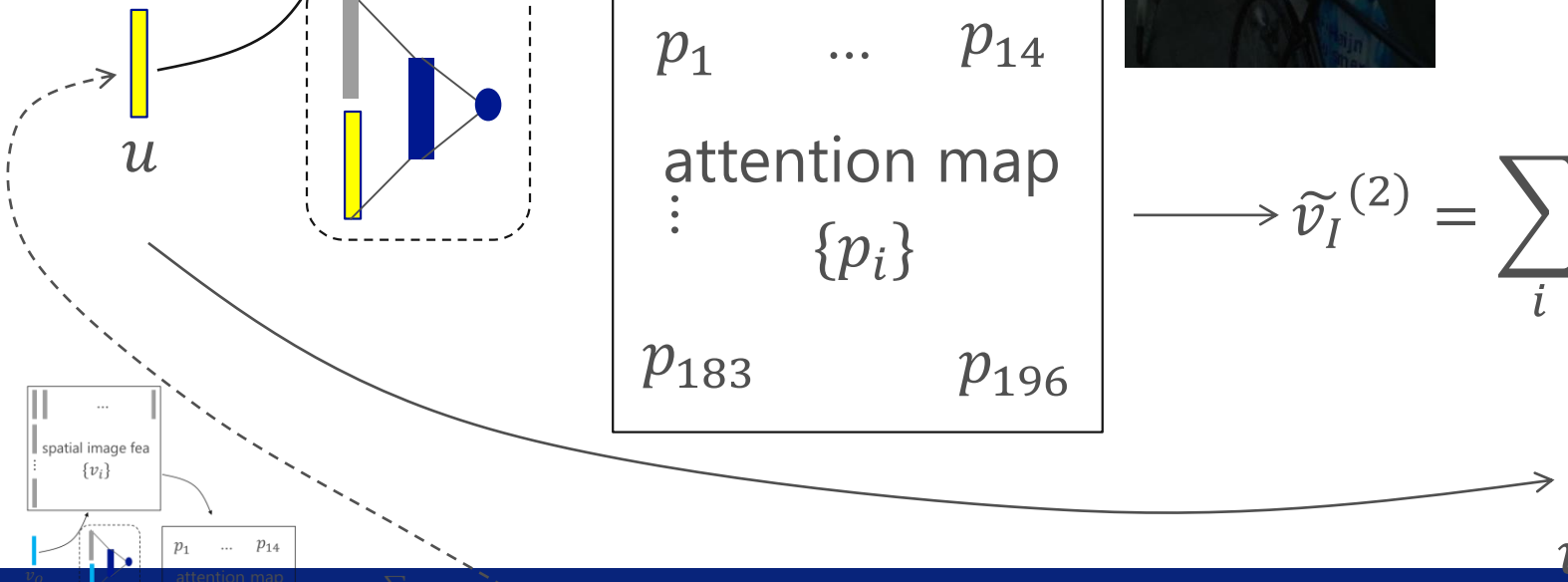


3. SAN: Computing the 1st level attention

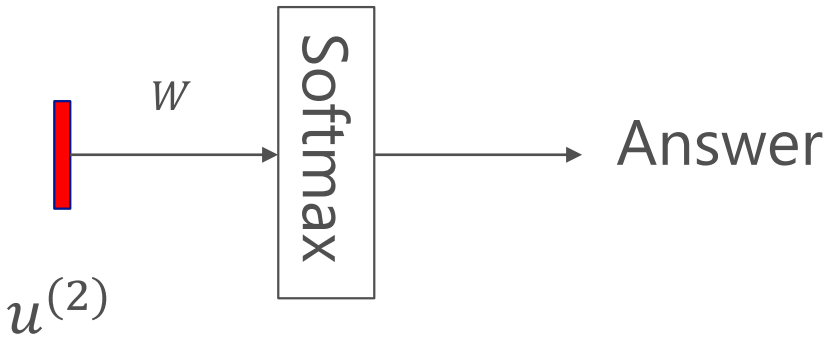


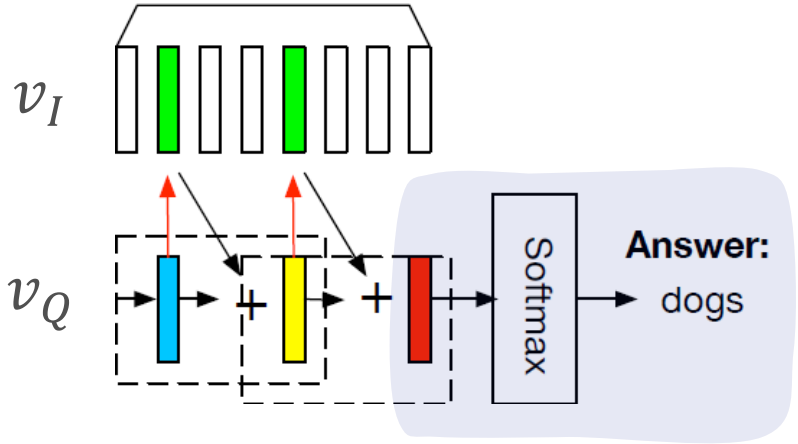
3. SAN: Compute the 2nd level attention





4. Answer prediction





$$p_{ans} = softmax(W\, u^{(2)} + b)$$

$$ans^* = \underset{\{ans\}}{\text{argmax}}\{p_{ans}\}$$

Results

Methods	test-dev All	test-dev Yes/No	test-dev Number	test-dev Other	test-std All
VQA: [1]					
Question	48.1	75.7	36.7	27.1	-
Image	28.1	64.0	0.4	3.8	-
Q+I	52.6	75.6	33.7	37.4	-
LSTM Q	48.8	78.2	35.7	26.6	-
LSTM Q+I	53.7	78.9	35.2	36.4	54.1
SAN(2, CNN)	**58.7**	**79.3**	**36.6**	**46.1**	**58.9**

Other:
Object
Color
Location
...

Table 5: VQA results on the official server, in percentage

Big improvement on the VQA benchmark (and COCO-QA, DAQUAR)
Improvement is mainly in the *Other* category.

Q: what stands between two blue lounge chairs on an empty beach?

1st attention layer

2nd attention layer

Answer: **umbrella**

Interim summary

Learn Sent2Vec by the DSSM (Open Source: http://aka.ms/sent2vec/)

- The DSSM projects the whole-sentence to a continuous space
- The DSSM is built on the character level
- The DSSM directly optimizes semantic similarity objective functions

Reinforcement learning for NLP tasks in a continuous space

- Project both states and actions (defined by *unbounded* NL) to a continuous semantic space using deep neural nets
- Compute the Q function in the continuous semantic space

Vision & language joint representation learning

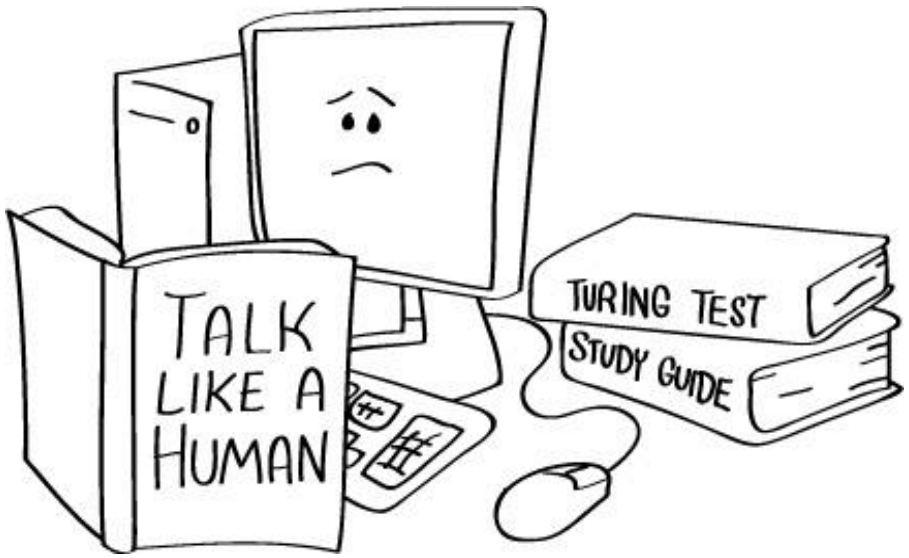
- Image Captioning – CaptionBot (http://CaptionBot.ai)
- Visual question answering – reasoning is the key challenge

Part IV

Natural Language Understanding

Natural Language Understanding

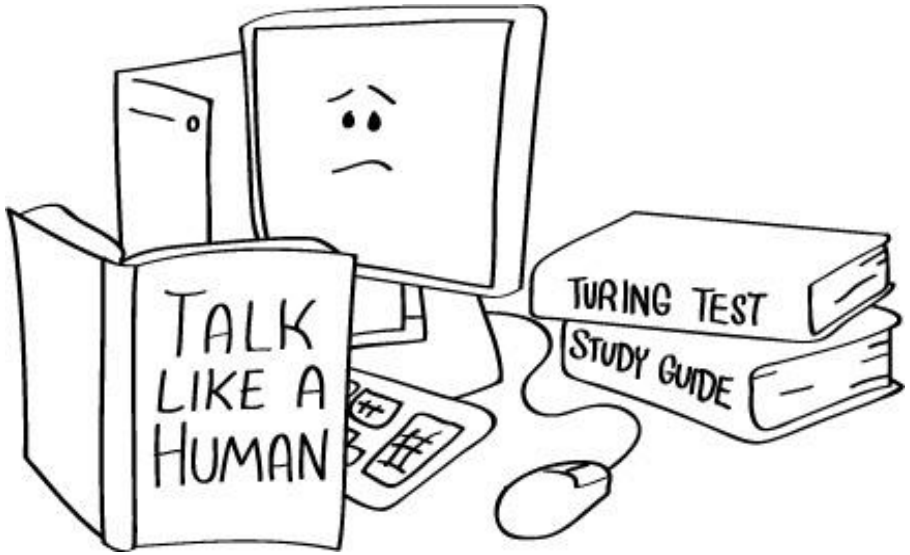
- Build an intelligent system that can interact with human using natural language
- Research challenge
  - Meaning representation of text
  - Support useful inferential tasks



http://csunplugged.org/turing-test

Natural Language Understanding

- Continuous Word Representations
  - Language is compositional
  - Word is the basic semantic unit
- Knowledge Base Embedding
- KB-based Question Answering & Machine Comprehension



http://csunplugged.org/turing-test

Continuous Word Representations

- A lot of popular methods for creating word vectors!
  - Vector Space Model [Salton & McGill 83]
  - Latent Semantic Analysis [Deerwester+ 90]
  - Brown Clustering [Brown+ 92]
  - Latent Dirichlet Allocation [Blei+ 01]
  - Deep Neural Networks [Collobert & Weston 08]
  - Word2Vec [Mikolov+ 13]
  - GloVe [Pennington+ 14]
- Encode term co-occurrence information
- Measure semantic similarity well

Semantic Embedding

Project raw text into a continuous semantic space

e.g., word embedding

Captures the word meaning in a semantic space

$f(\boldsymbol{cat}) =$

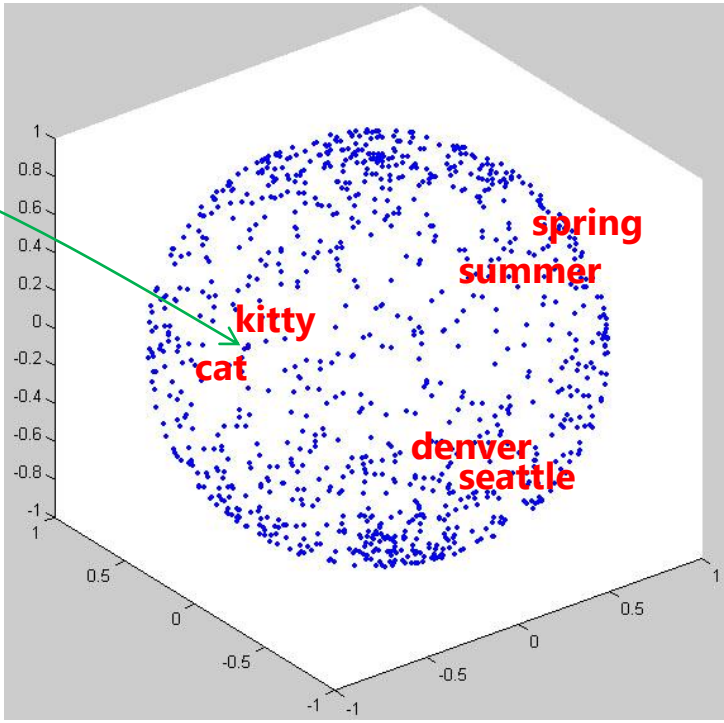
The index of "cat" in the vocabulary

a.k.a the 1-hot word vector

word embedding vector in the semantic space

Dim=100~1000

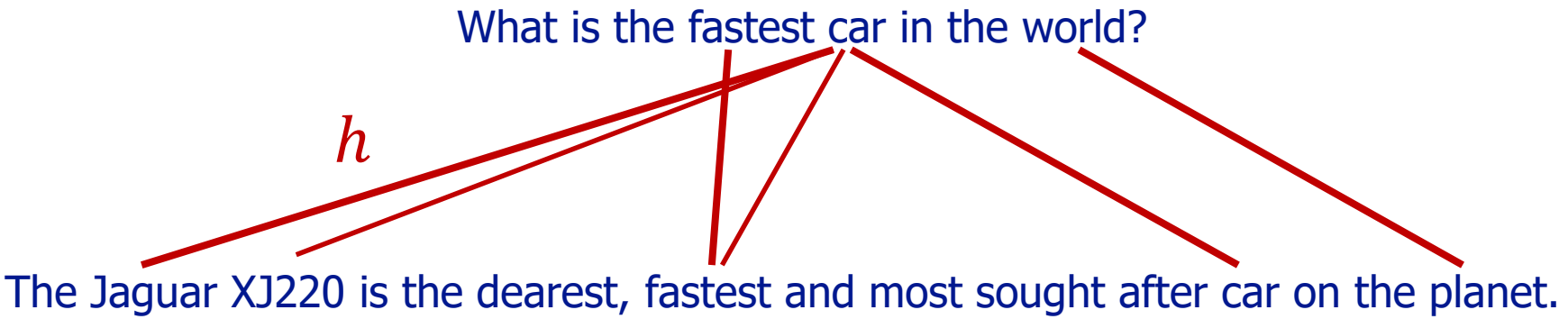
Dim=|V|=100K~100M



Deerwester, Dumais, Furnas, Landauer, Harshman, "Indexing by latent semantic analysis," JASIS 1990

Why is Word Embedding Useful?

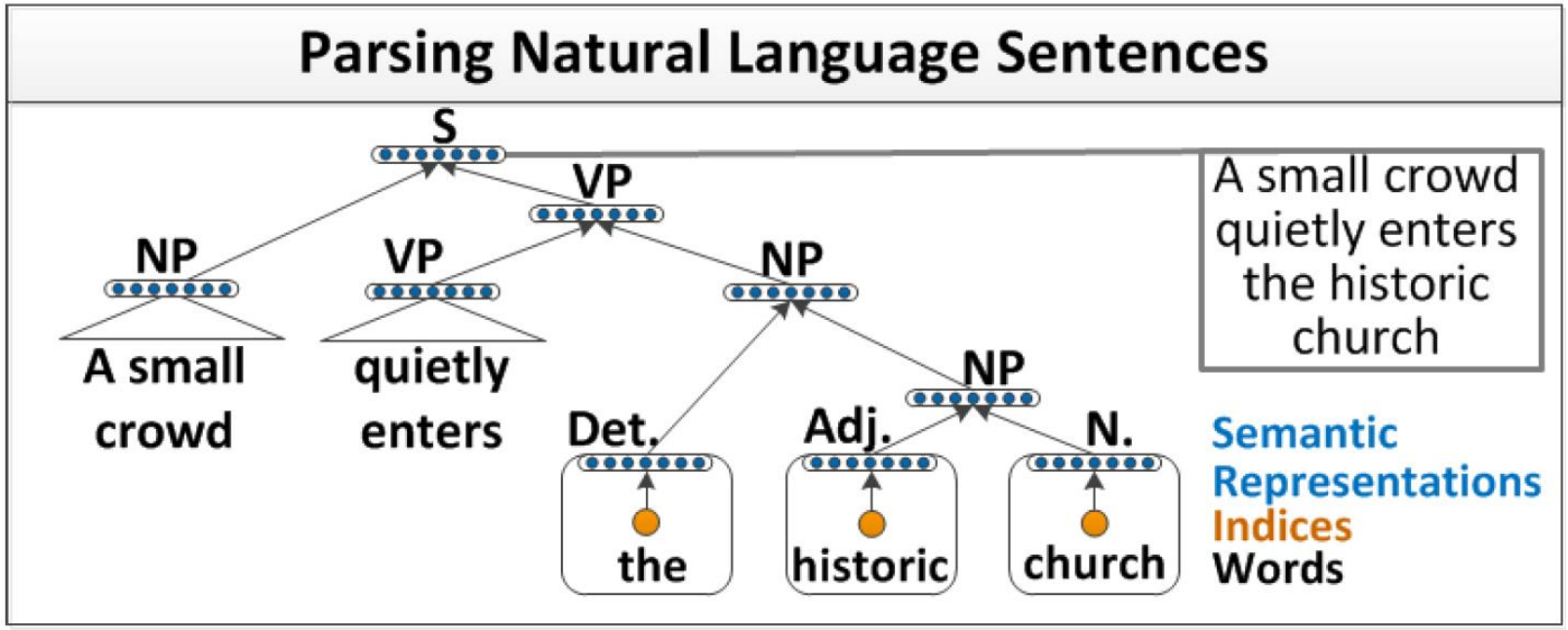
- Lexical semantics – semantic word similarity
  - Used as features in many NLP applications
  - e.g., Question/Sentence matching [Yih+ ACL-13; Jansen+ ACL-14]



- Simple semantic representation of text
  - Represent longer text using average of the word vectors
  - e.g., entity [Socher+ NIPS-13], question [Berant&Liang ACL-14]

Why is Word Embedding Useful? (Cont'd)

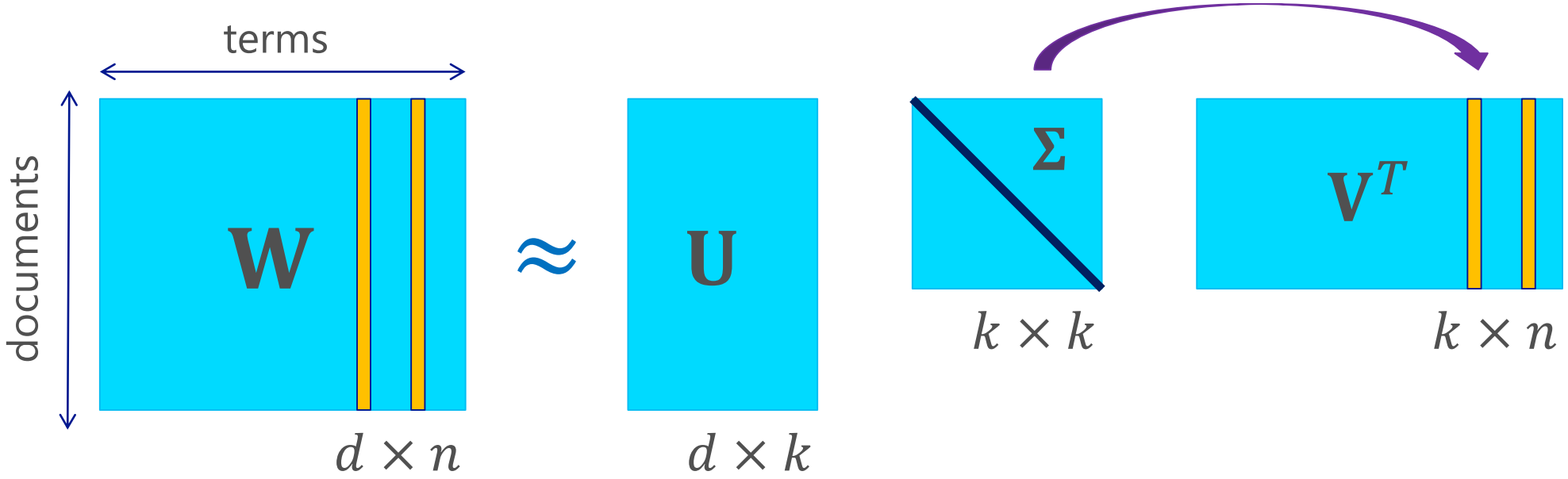
- "Pre-training" of a neural-network model
  - Take word vectors trained on a general corpus as input
  - e.g., Recursive NN for parsing [Socher+ ICML-11]



Roadmap – Continuous Word Representations

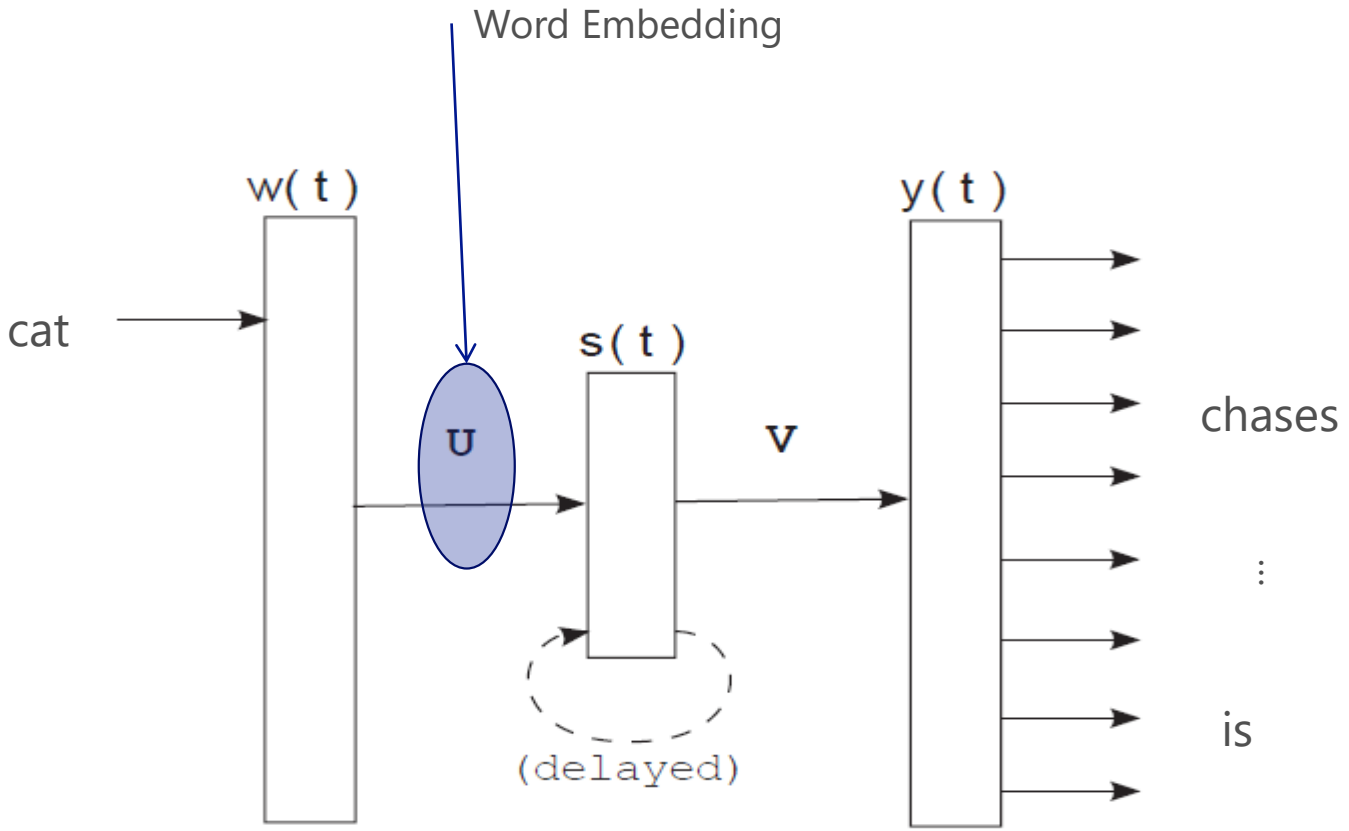
- Samples of word embedding models
  - Latent Semantic Analysis (LSA), Recurrent Neural Networks
  - SENNA, CBOW/Skip-gram, DSSM, GloVe
- Evaluation
  - Semantic word similarity
  - Relational similarity (word analogy)
- Related work
  - Model different word relations
  - Other word embedding models

Latent Semantic Analysis



- SVD generalizes the original data
- Uncovers relationships not explicit in the thesaurus
- Term vectors projected to k-dim latent space
- Word similarity: cosine of two column vectors in $\mathbf{\Sigma V}^T$

RNN-LM Word Embedding



Mikolov, Yih, Zweig, "Linguistic Regularities in Continuous Space Word Representations," NAACL 2013

SENNA Word Embedding

Scoring:

$$Score(w_1, w_2, w_3, w_4, w_5) = U^T \sigma(W[f_1, f_2, f_3, f_4, f_5] + b)$$

Training:

$$J = \max(0, 1 + S^- - S^+)$$

Update the model until $S^+ > 1 + S^-$

Where

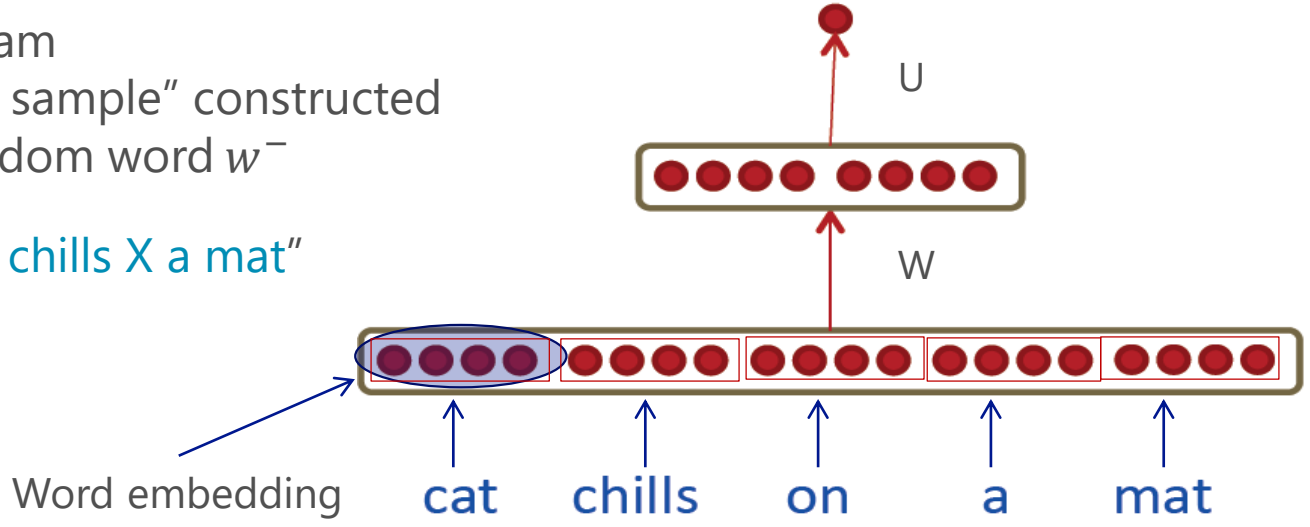
$$S^+ = Score(w_1, w_2, w_3, w_4, w_5)$$
$$S^- = Score(w_1, w_2, w^-, w_4, w_5)$$

And

$< w_1, w_2, w_3, w_4, w_5 >$ is a valid 5-gram

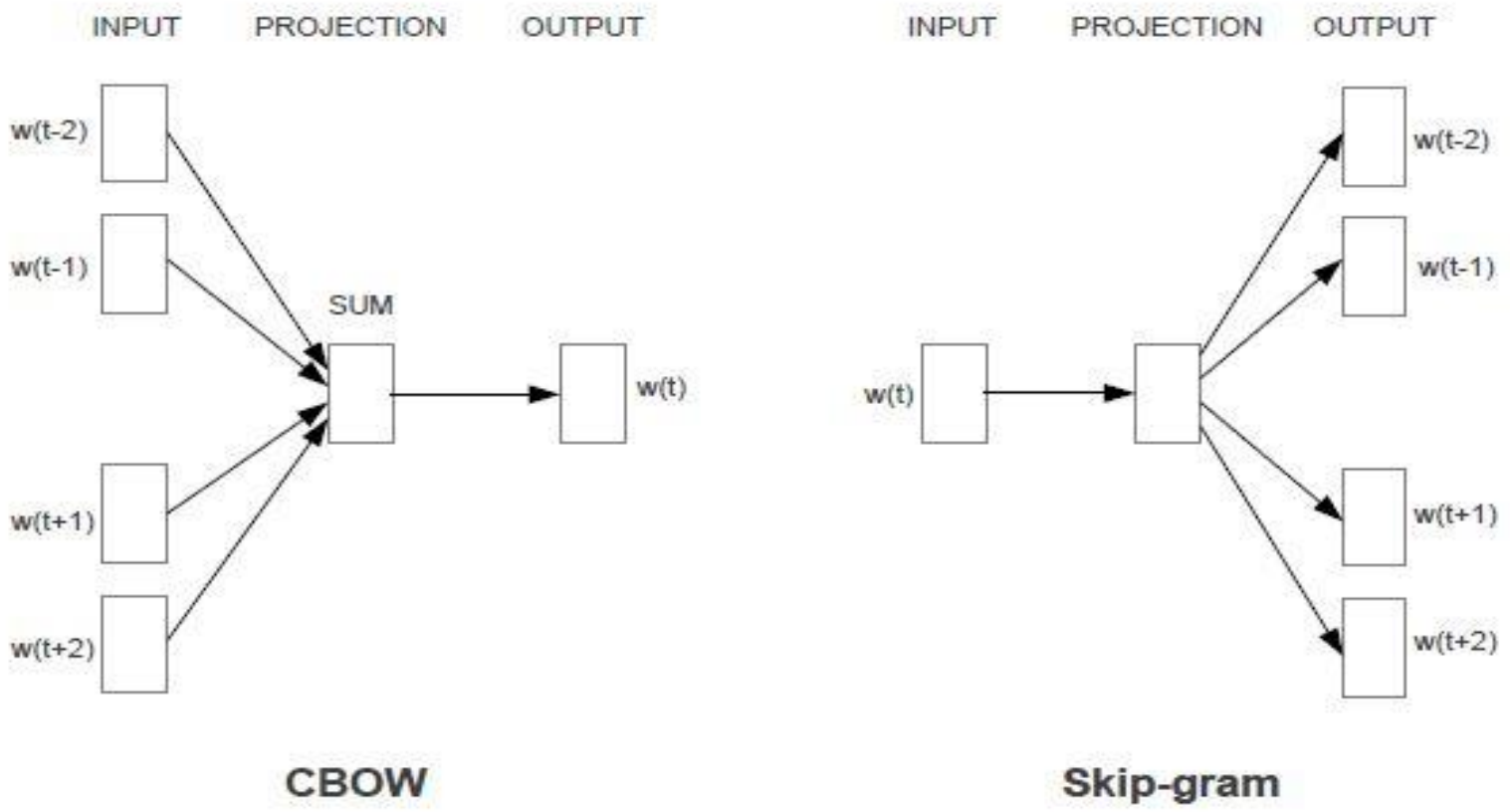
$< w_1, w_2, w^-, w_4, w_5 >$ is a "negative sample" constructed by replacing the word w_3 with a random word w^-

e.g., a negative example: "cat chills X a mat"



Collobert, Weston, Bottou, Karlen, Kavukcuoglu, Kuksa, "Natural Language Processing (Almost) from Scratch," JMLR 2011

CBOW/Skip-gram Word Embeddings



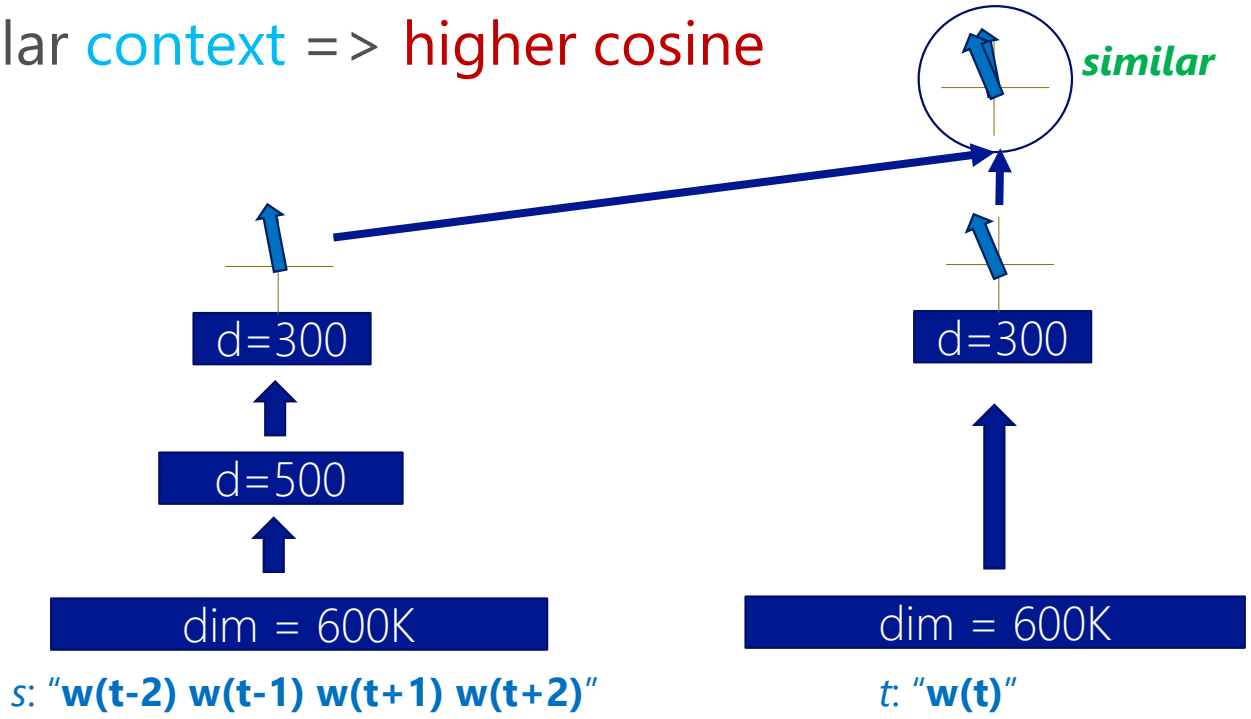
Continuous Bag-of-Words

The CBOW architecture (a) on the left, and the Skip-gram architecture (b) on the right. [Mikolov et al., 2013 ICLR].

DSSM: Learning Word Meaning

- Learn a word's semantic meaning by means of its neighbors (context)
  - Construct context <-> word training pair for DSSM
  - Similar words with similar context => higher cosine
- Training Condition:
  - 600K vocabulary size
  - 1B words from Wikipedia
  - 300-dimentional vector

You shall know a word by the company it keeps
(J. R. Firth 1957: 11)



[Song, He, Gao, Deng, 2014]

GloVe: Global Vectors for Word Representation [Pennington+ EMNLP-14]

- Semantic relatedness can be observed from word co-occurrence counts and ratios

Context words

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

"solid" is more related to "ice"

GloVe: Global Vectors for Word Representation [Pennington+ EMNLP-14]

- Semantic relatedness can be observed from word co-occurrence counts and ratios

Context words

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

"gas" is more related to "steam"

GloVe: Global Vectors for Word Representation [Pennington+ EMNLP-14]

- Semantic relatedness can be observed from word co-occurrence counts and ratios

Context words

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

Equally related or unrelated

GloVe: Global Vectors for Word Representation [Pennington+ EMNLP-14]

- Word embedding model design principle:

$$F(w_i, w_j, \tilde{w}_k) = \frac{P(k|i)}{P(k|j)}$$ (e.g., i = ice, j = steam, k = solid/gas)

- Objective: $$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2$$

Down weight low co-occurrences

co-occurrence counts

Evaluation: Semantic Word Similarity

- Data: word pairs with human judgment (e.g., WS-353, RG-65)

Word 1	Word 2	Human Score (mean)
midday	noon	9.3
tiger	jaguar	8.0
cup	food	5.0
forest	graveyard	1.9
…	…	…

- Correlation of the *ranking* of word similarity and human judgment
  - **Spearman's rank correlation coefficient** ρ
- Word embedding models individually usually do not achieve the state-of-the-art results (cf. ACL Wiki Similarity (State-of-the-art))

Evaluation: Relational Similarity (Word Analogy)

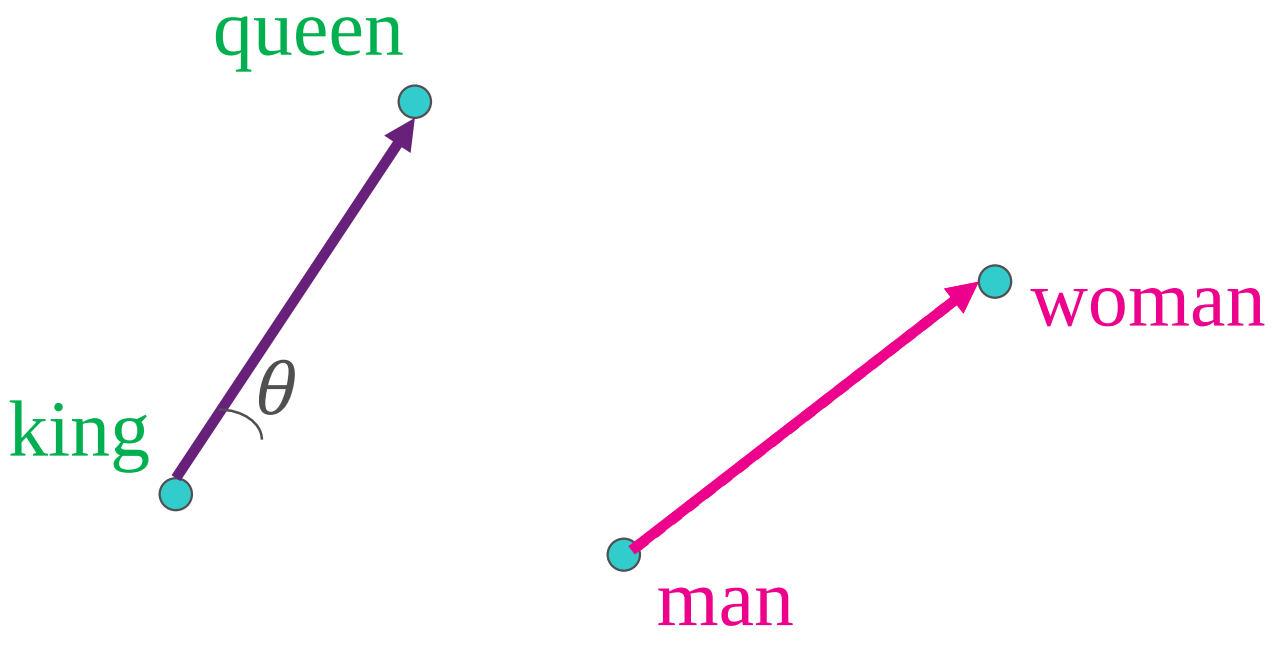
king : queen $\stackrel{?}{=}$ man : woman

- Determine whether two pairs of words have the same relation (the "analogy" problem) [Bejar et al. '91]
  - (silverware : fork) vs. (clothing : shirt) [singular collective]
  - (coast : ocean) vs. (sidewalk : road) [contiguity]
  - (psychology : mind) vs. (astronomy : stars) [knowledge]
- Why it's useful?

  Building a general "relational similarity" model is a more efficient way to learn a model for any arbitrary relation [Turney, 2008]

Unexpected Finding: Directional Similarity

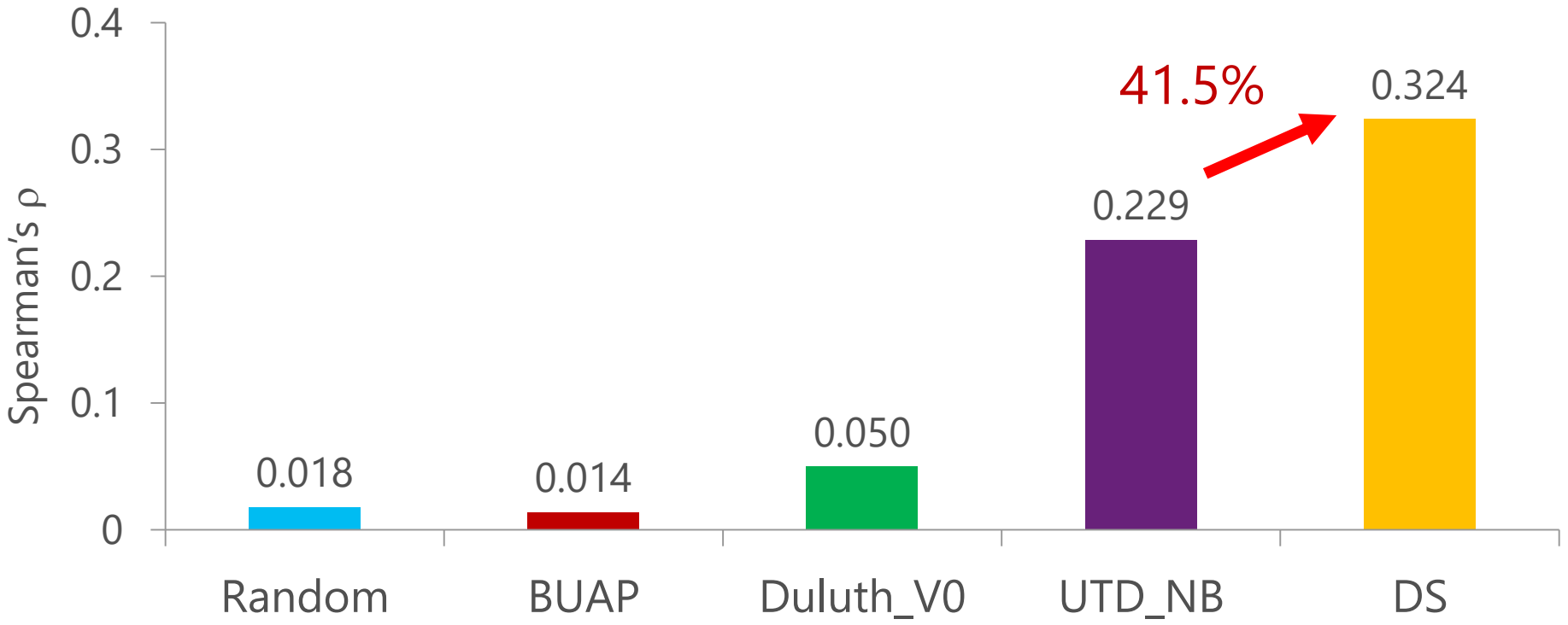
- Word embedding taken from recurrent neural network language model (RNN-LM) [Mikolov+ 2011]



- Relational similarity is derived by the cosine score

Experimental Results

- SemEval-2012 Task 2 – Relational Similarity
  - Rank word pairs of 69 testing relations
  - Evaluate model by its correlation to human judgments



Similar Results Observed on Other Datasets

- MSR syntactic test set [Mikolov+ 2013]
  - see : saw = return : returned
  - better : best = rough : roughest

- Semantic-Syntactic word relationship [Mikolov+ 2013]
  - Athens : Greece = Oslo : Norway
  - brother : sister = grandson : granddaughter
  - apparent : apparently = rapid : rapidly

Evaluation on Word Analogy

The dataset contains 19,544 word analogy questions:

- Semantic questions, e.g.,: "Athens is to Greece as Berlin is to ?"
- Syntactic questions, e.g.,: "dance is to dancing as fly is to ?"

Model	Dim	Size	Accuracy Avg.(sem+syn)
SG	300	1B	61.0%
CBOW	300	1.6B	36.1%
vLBL	300	1.5B	60.0%
ivLBL	300	1.5B	64.0%
GloVe	300	1.6B	70.3%
DSSM	300	1B	71.9%

(i)vLBL from (Mnih et al., 2013); skip-gram (SG) and CBOW from (Mikolov et al., 2013a,b); GloVe from (Pennington+, 2014)

Discussion

- Directional Similarity cannot handle symmetric relations
  - **good : bad = bad : good**
- Vector arithmetic = Similarity arithmetic
  [Levy & Goldberg CoNLL-14]
- Find the closest x to $king - man + woman$ by

$$\arg\max_x(\cos(x, king - man + woman)) =$$
$$\arg\max_x(\cos(x, king) - \cos(x, man) + \cos(x, woman))$$

Related Work – Model Different Word Relations

Tomorrow will be rainy.

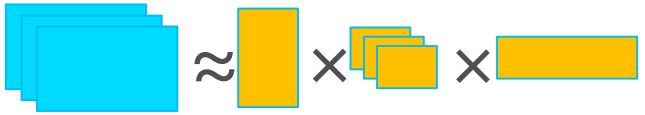
Tomorrow will be sunny.

$similar$(r , sunny)?

$antonym$(rainy, sunny)?

- Multi-Relational Latent Semantic Analysis [Chang+ EMNLP-04]

f_{rel}(,)

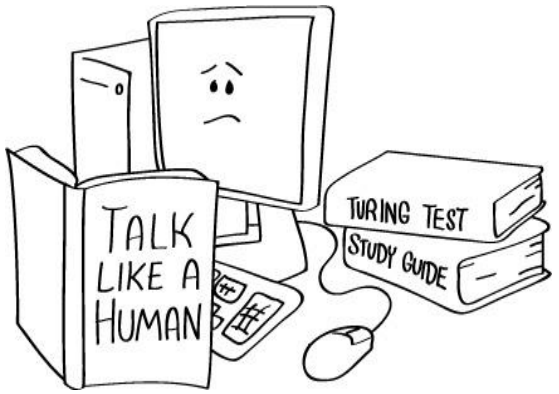


Related Work – Word Embedding Models

- Other word embedding models
  - [Wang+ EMNLP-14], [Bian+ ECML/PKDD-14], [Xu+, CIKM-14], [Faruqui+ NAACL-15], [Yogatama+ ICML-15], [Faruqui+ ACL-15]
- Analysis of Word2Vec and Directional Similarity
  - **Linguistic Regularities in Sparse and Explicit Word Representations** [Levy & Goldberg CoNLL-14]
  - **Neural Word Embedding as Implicit Matrix Factorization** [Levy & Goldberg NIPS-14]
- Theoretical justification and unification
  - **Word Embeddings as Metric Recovery in Semantic Spaces** [Hashimoto+ TACL-16]
- New Evaluation: RelEval@ACL-16 – Evaluating Vector Space Representations for NLP

Natural Language Understanding

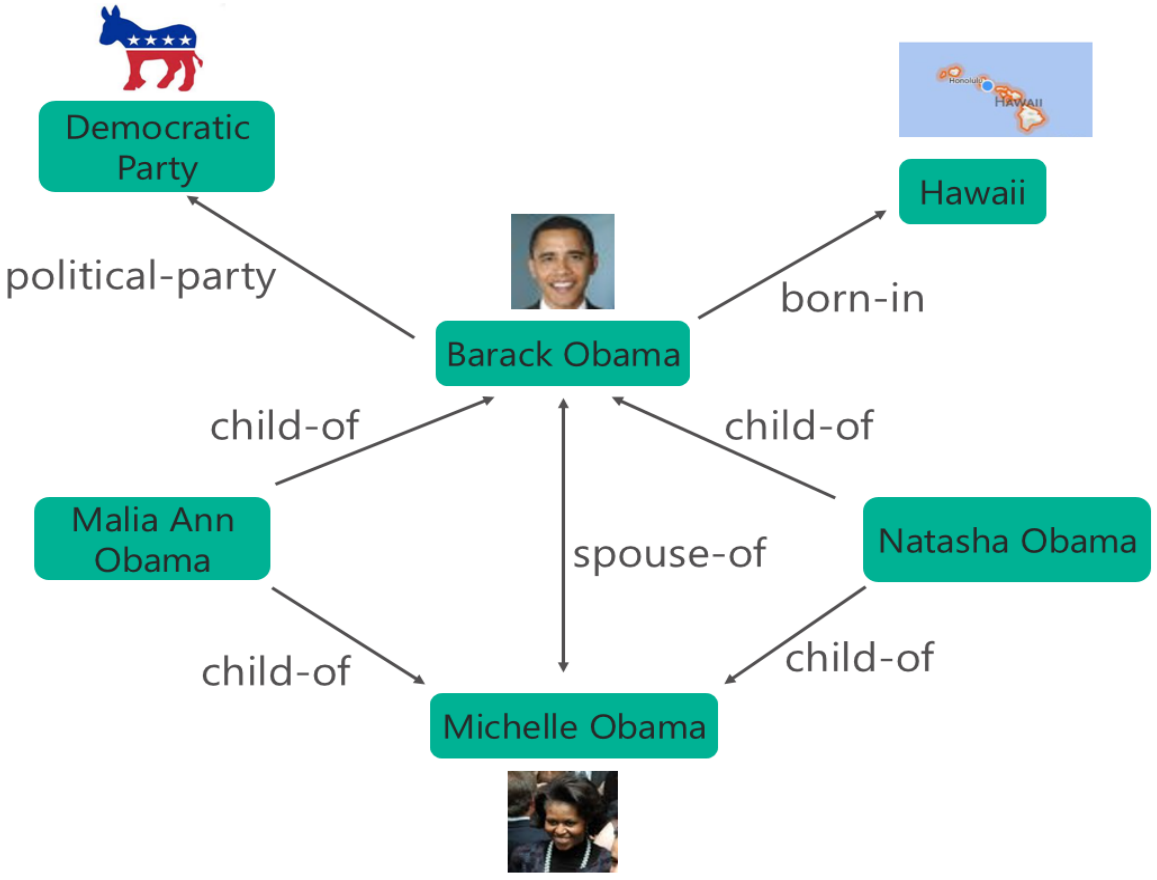
- Continuous Word Representations & Lexical Semantics
- Knowledge Base Embedding
  - Nickel et al., "A Review of Relational Machine Learning for Knowledge Graphs"
- KB-based Question Answering & Machine Comprehension

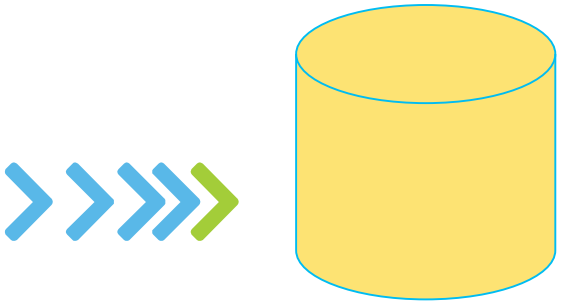


http://csunplugged.org/turing-test

Knowledge Base

- Captures world knowledge by storing properties of millions of entities, as well as relations among them





Freebase
DBpedia
YAGO
NELL
OpenIE/ReVerb

Current KB Applications in NLP & IR

- Question Answering

  "*What are the names of Obama's daughters?*"

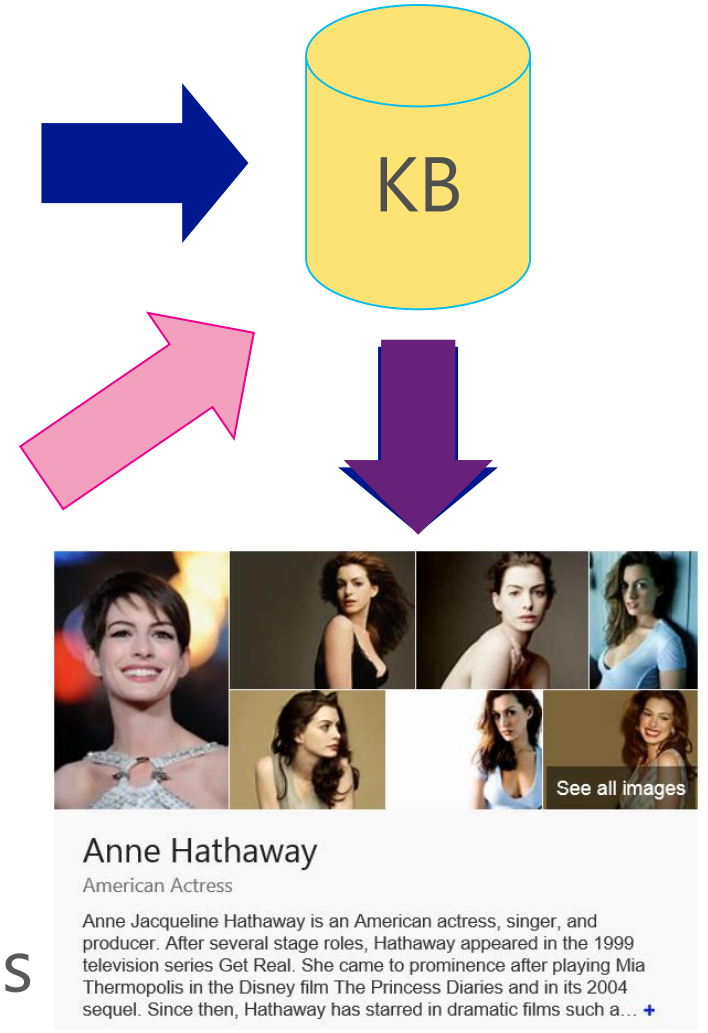
  $\lambda x.parent(Obama, x) \wedge gender(x, Female)$

- Information Extraction
  - "*Hathaway was born in Brooklyn, New York.*"

  $bornIn(Hathaway, Brooklyn)$

  $contains(New\ York, Brooklyn)$

- Web Search
  - Identify entities and relationships in queries



Reasoning with Knowledge Base

- Knowledge base is never complete!
  - Predict new facts: $Nationality(Natasha\ Obama, ?)$
  - Mine rules: $BornInCity(a, b) \wedge CityInCountry(b, c) \Rightarrow Nationality(a, c)$
- Modeling multi-relational data
  - Statistical relational learning [Getoor & Taskar, 2007]
  - Path ranking methods (e.g., random walk) [e.g., Lao+ 2011]
  - Knowledge base embedding
    - Very efficient
    - Better prediction accuracy

Knowledge Base Embedding

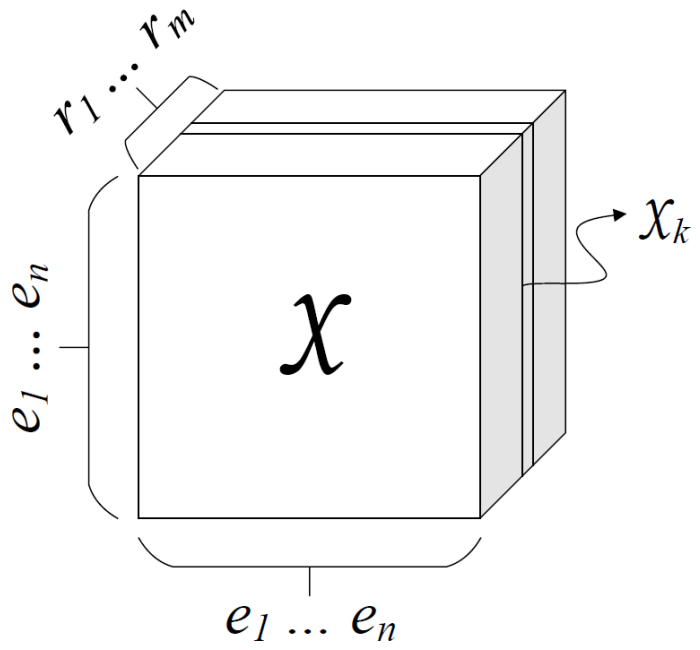
- Each entity in a KB is represented by an $\boldsymbol{R}^{\boldsymbol{d}}$ vector
- Predict whether $(\boldsymbol{e_1}, \boldsymbol{r}, \boldsymbol{e_2})$ is true by $\boldsymbol{f_r}(\boldsymbol{v_{e_1}}, \boldsymbol{v_{e_2}})$

- Recent work on KB embedding
  - Tensor decomposition
    - RESCAL [Nickel+, ICML-11], TRESCAL [Chang+, EMNLP-14]
  - Neural networks
    - SME [Bordes+, AISTATS-12], NTN [Socher+, NIPS-13], TransE [Bordes+, NIPS-13]

Tensor Decomposition: Knowledge Base Representation (1/2)

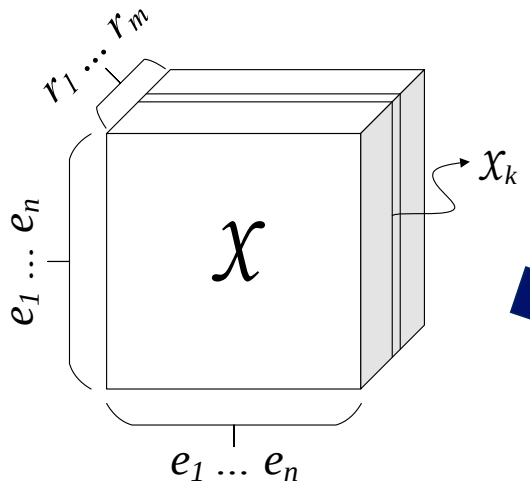
- Collection of **subj-pred-obj** triples – (e_1, r, e_2)

Subject	Predicate	Object
Obama	BornIn	Hawaii
Bill Gates	Nationality	USA
Bill Clinton	SpouseOf	Hillary Clinton
Satya Nadella	WorkAt	Microsoft
...	...	...

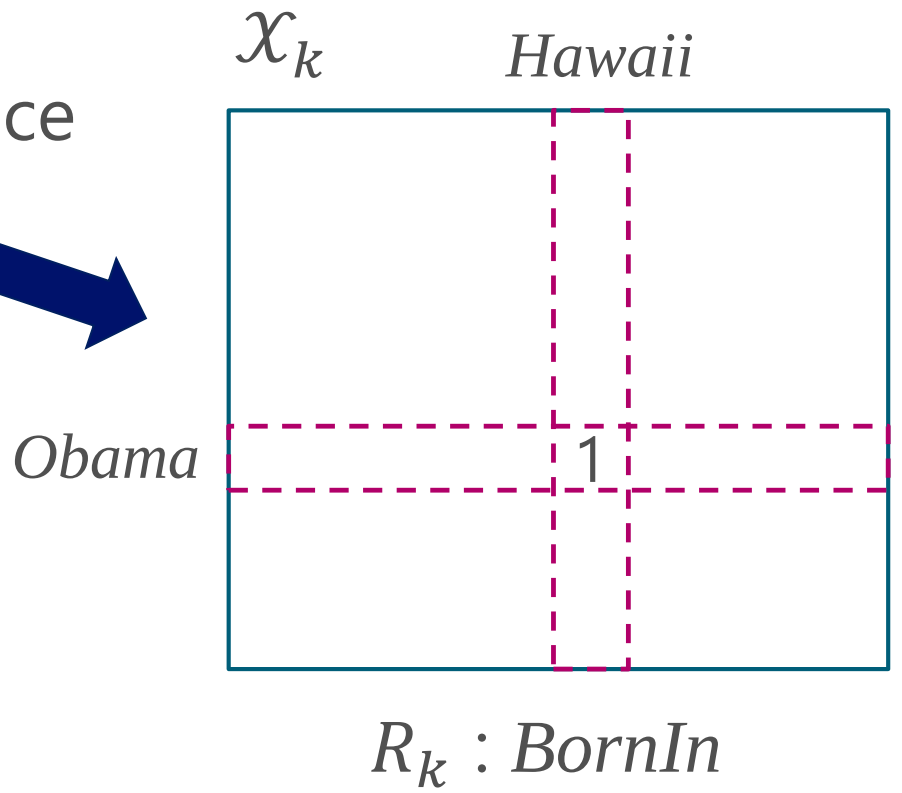


n: # entities, m: # relations

Tensor Decomposition: Knowledge Base Representation (2/2)



k-th slice

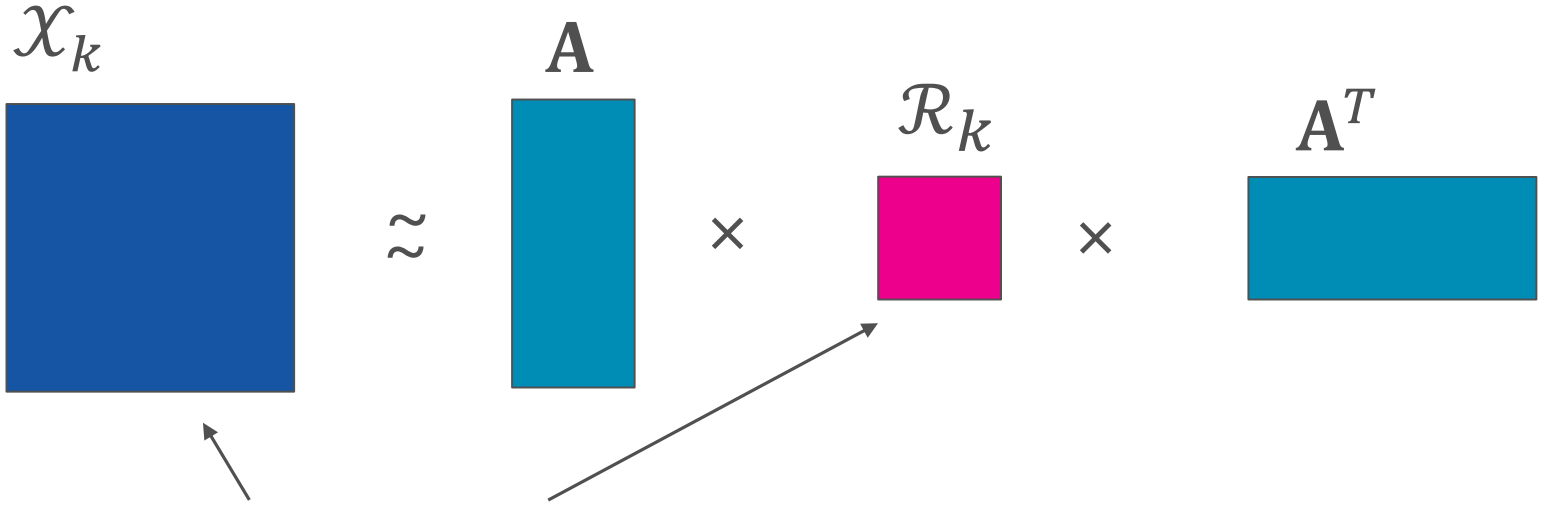


A zero entry means either:

- Incorrect (*false*)
- Unknown

Tensor Decomposition Objective

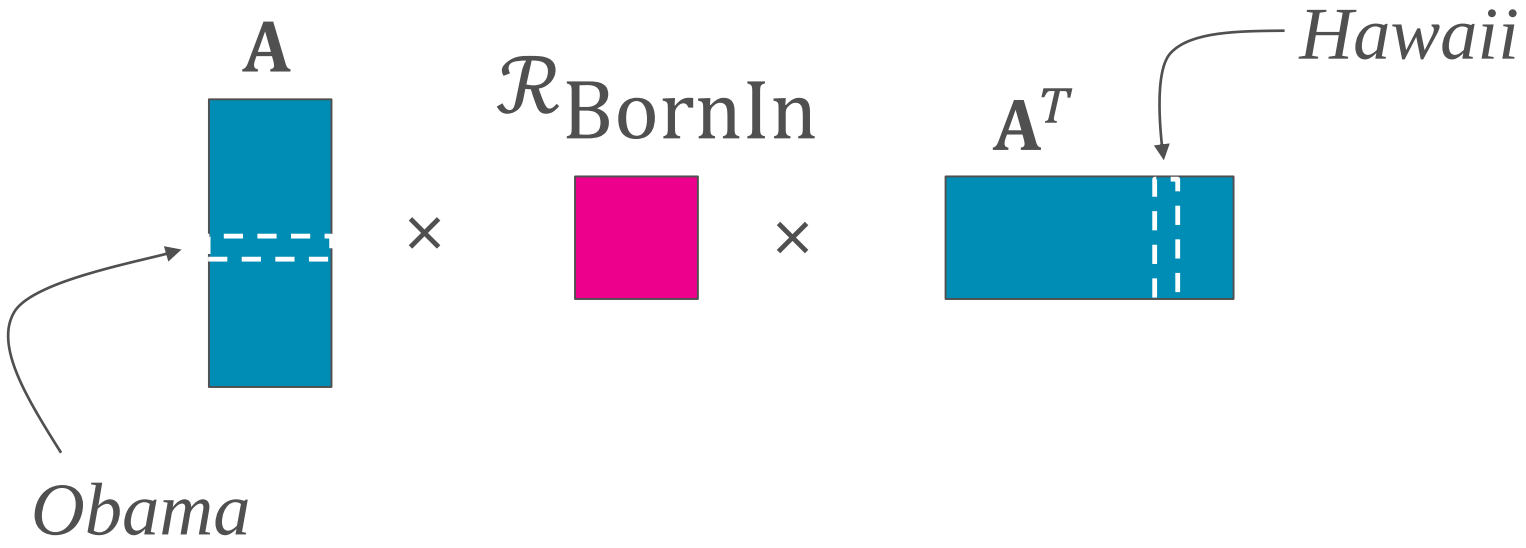
- Objective: $\frac{1}{2}\underbrace{\left(\sum_k \|\mathcal{X}_k - \mathbf{A}\mathcal{R}_k\mathbf{A}^T\|_F^2\right)}_{\textit{Reconstruction Error}} + \frac{1}{2}\underbrace{\left(\|A\|_F^2 + \sum_k \|\mathcal{R}_k\|_F^2\right)}_{\textit{Regularization}}$



RESCAL [Nickel+, ICML-11]

Measure the Degree of a Relationship

$$f_{\text{BornIn}}(\text{Obama}, \text{Hawaii}) = \mathbf{A}_{\text{Obama},:}\, \mathcal{R}_{\text{BornIn}}\, \mathbf{A}^{\text{T}}_{\text{Hawaii},:}$$

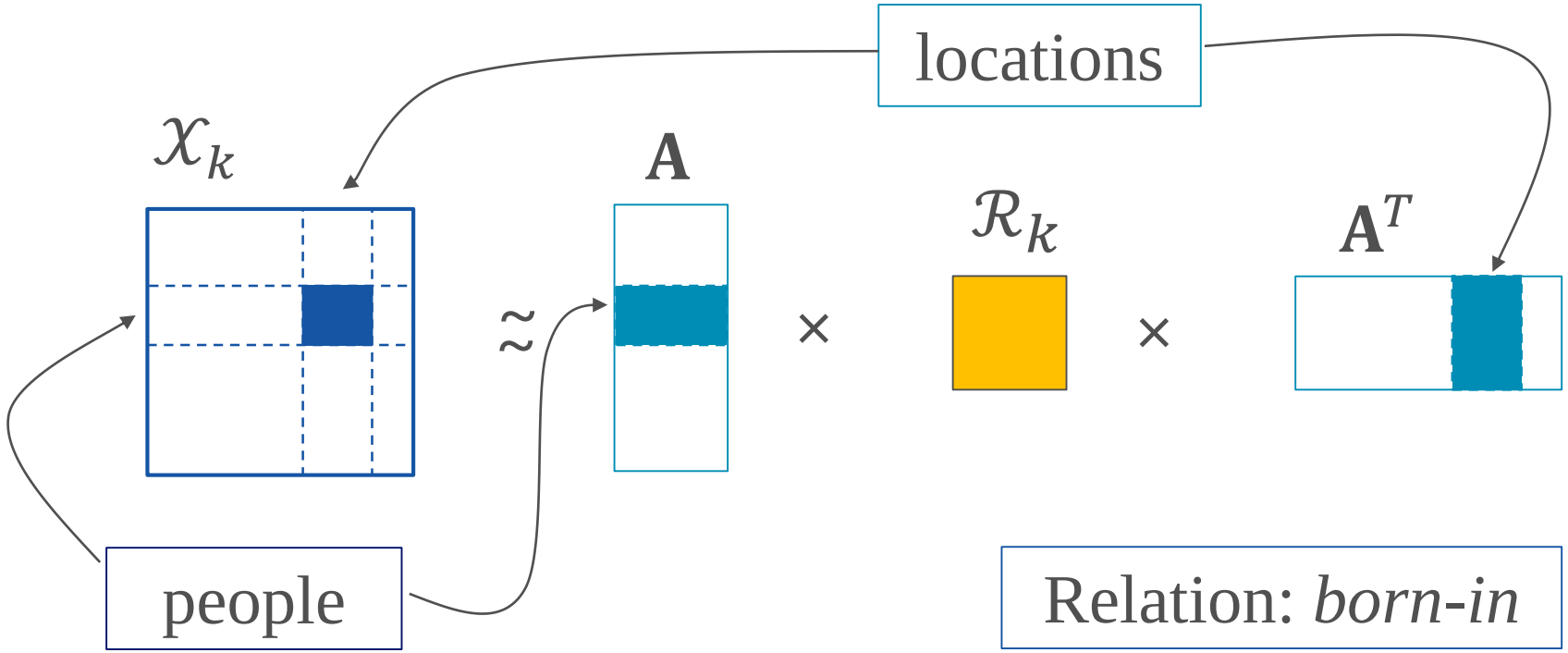


Typed Tensor Decomposition – TRESCAL
[Chang+ EMNLP-14]

- Relational domain knowledge
  - Type information and constraints
  - Only legitimate entities are included in the loss
- Benefits of leveraging type information
  - Faster model training time
  - Highly scalable to large KB
  - Higher prediction accuracy

Typed Tensor Decomposition Objective

- Reconstruction error: $\frac{1}{2}\sum_k \|\mathcal{X}_k - \mathbf{A}\mathcal{R}_k\mathbf{A}^T\|_F^2$



Typed Tensor Decomposition Objective

- Reconstruction error: $\frac{1}{2}\sum_k \|\mathcal{X}'_k - \mathbf{A}_{k_l}\mathcal{R}_k\mathbf{A}_{k_r}^T\|_F^2$

Training Procedure – Alternating Least-Squares (ALS) Method

Fix $\mathcal{R}_k$, update $\mathbf{A}$

Fix $\mathbf{A}$, update $\mathcal{R}_k$

Training Procedure – Alternating Least-Squares (ALS) Method

$$\mathbf{A} \leftarrow \left[\sum_k \mathcal{X}'_k \mathbf{A}_{k_r} \mathcal{R}_k^{\mathrm{T}} + \mathcal{X}'^{\mathrm{T}}_k \mathbf{A}_{k_l} \mathcal{R}_k\right] \left[\sum_k B_{k_r} + C_{k_l} + \lambda \mathbf{I}\right]^{-1}$$

where $B_{k_r} = \mathcal{R}_k \mathbf{A}_{k_r}^{\mathrm{T}} \mathbf{A}_{k_r} \mathcal{R}_k^{\mathrm{T}}$, $C_{k_l} = \mathcal{R}_k^{\mathrm{T}} \mathbf{A}_{k_l}^{\mathrm{T}} \mathbf{A}_{k_l} \mathcal{R}_k$.

$$\mathbf{vec}(\mathcal{R}_k) \leftarrow \left(\mathbf{A}_{k_r}^{\mathrm{T}} \mathbf{A}_{k_r} \otimes \mathbf{A}_{k_l}^{\mathrm{T}} \mathbf{A}_{k_l} + \lambda \mathbf{I}\right)^{-1} \times \mathbf{vec}\left(\mathbf{A}_{k_l}^{\mathrm{T}} \mathcal{X}'_k \mathbf{A}_{k_r}\right)$$

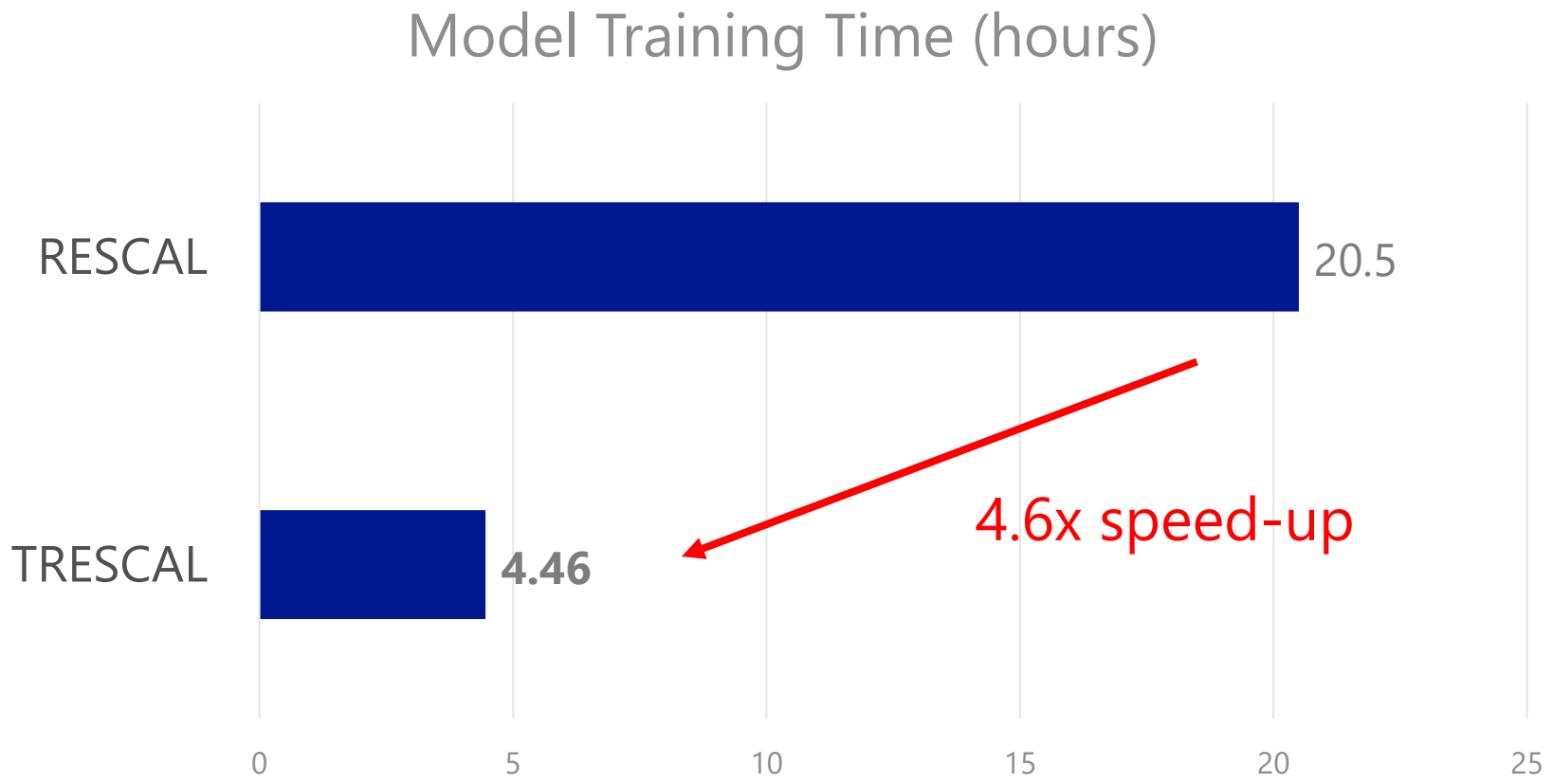
Experiments – KB Completion

- KB – Never Ending Language Learning (NELL)
  - Training: version 165
  - Developing: new facts between v.166 and v.533
  - Testing: new facts between v.534 and v.745

- Data statistics of the training set

# Entities	753k
# Relation Types	229
# Entity Types	300
# Entity-Relation Triples	1.8M

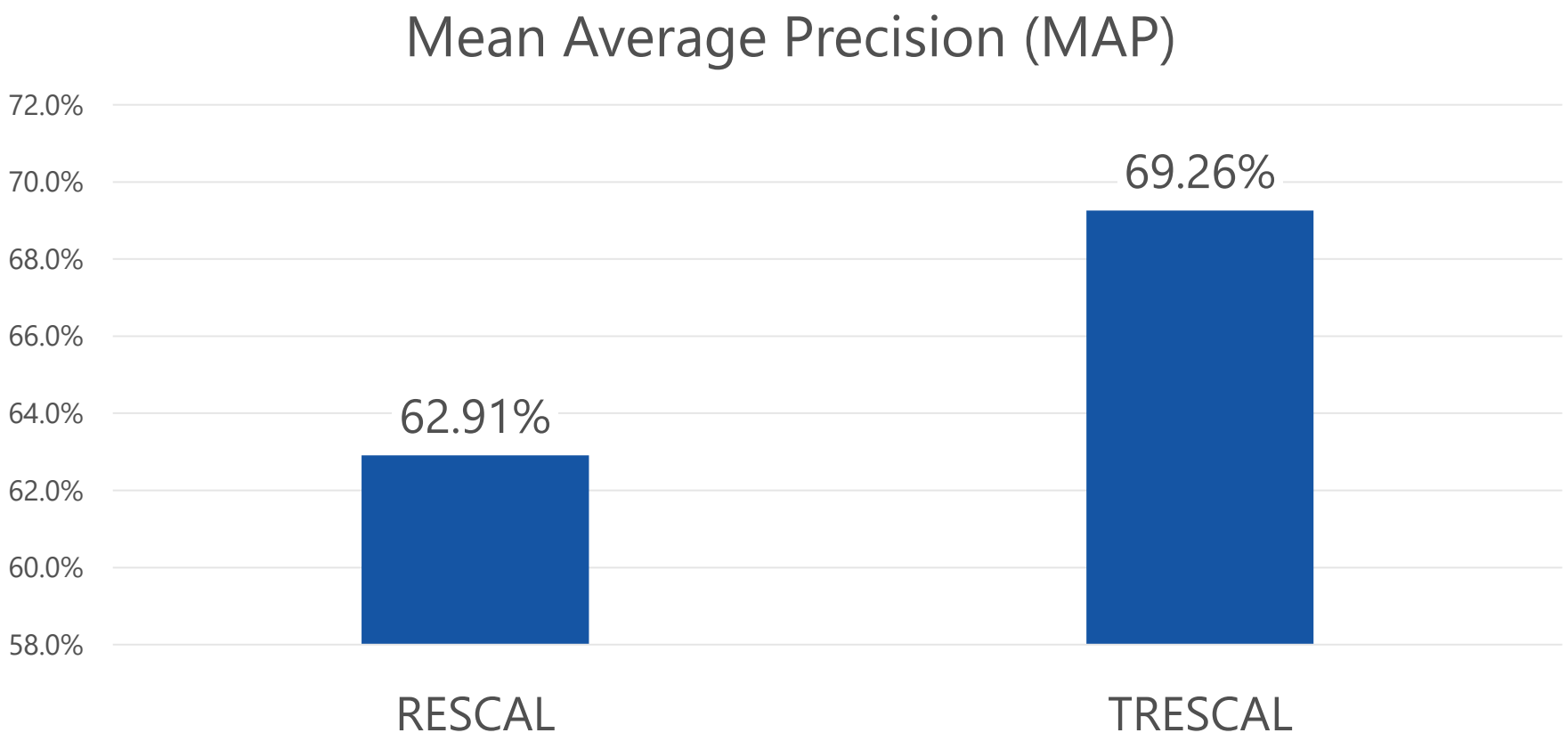
Training Time Reduction



- Both models finish training in 10 iterations.
- TRESCAL filters 96% entity triples with incompatible types.

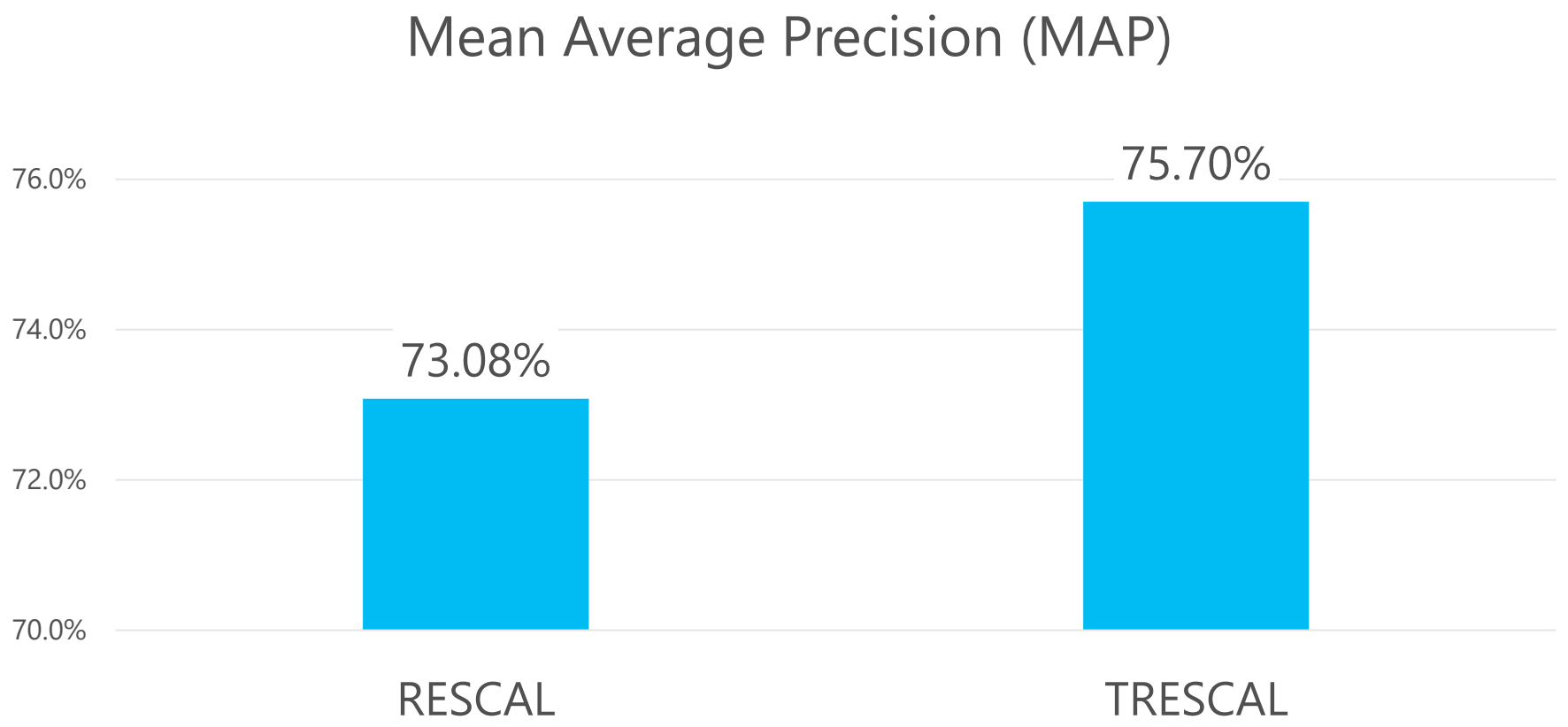
Entity Retrieval $(e_i, r_k, ?)$

- One positive entity with 100 negative entities

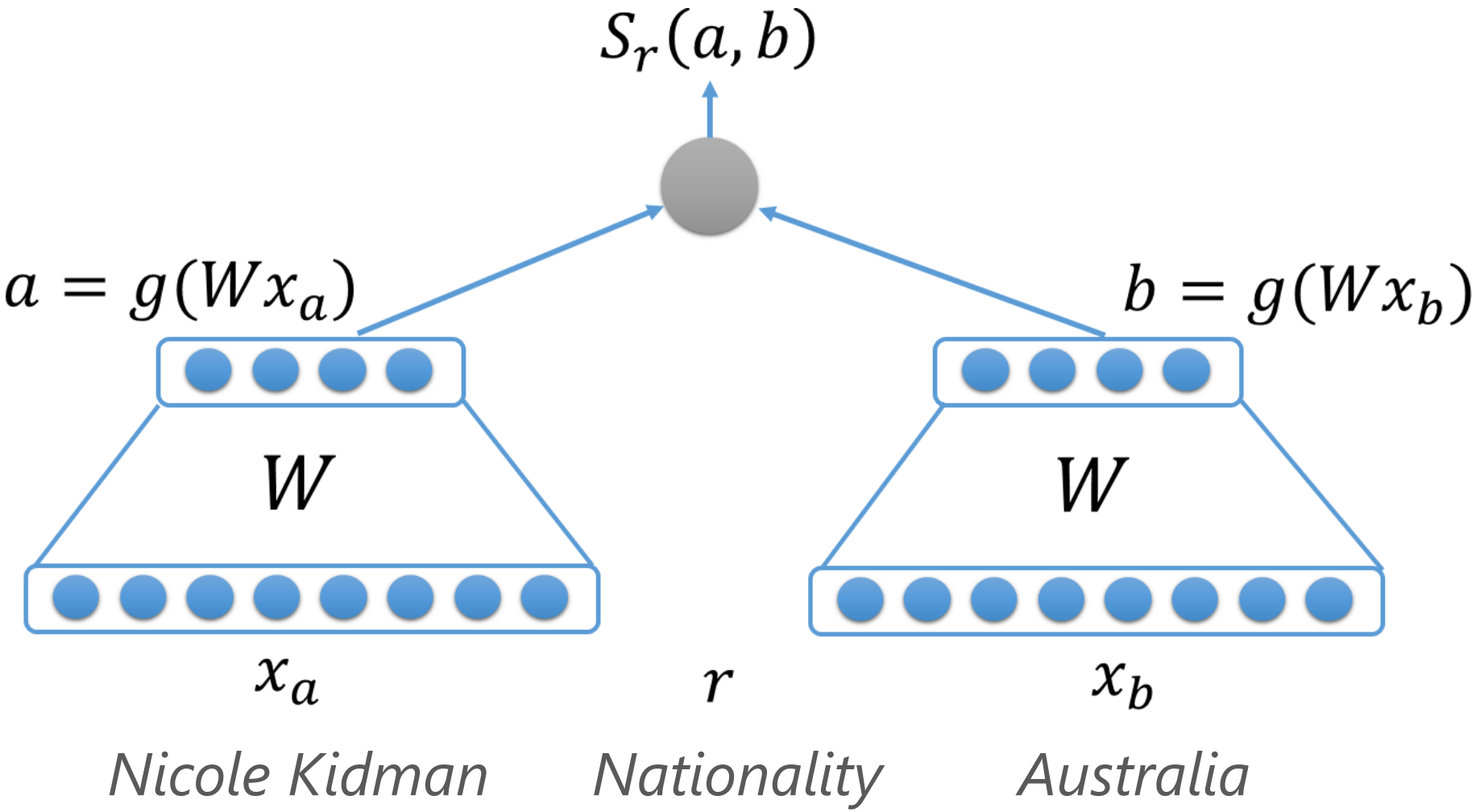


Relation Retrieval $(e_i, ?, e_j)$

- Positive entity pairs with equal number of negative pairs



Neural Knowledge Base Embedding



Relation Operators

Relation representation	Scoring Function $S_r(a, b)$	# Parameters
Vector (TransE) (Bordes+ 2013)	$\|\|a - b + V_r\|\|_{1,2}$	$O(n_r \times k)$
Matrix (Bilinear) (Bordes+ 2012, Collobert & Weston 2008)	$a^T M_r b$ $u^T f(M_{r1} a + M_{r2} b)$	$O(n_r \times k^2)$
Tensor (NTN) (Socher+ 2013)	$u^T f(a^T T_r b + M_{r1} a + M_{r2} b)$	$O(n_r \times k^2 \times d)$
Diagonal Matrix (Bilinear-Diag) (Yang+ 2015)	$a^T diag(M_r) b$	$O(n_r \times k)$

n_r: #predicates, k: #dimensions of entity vectors, d: #layers

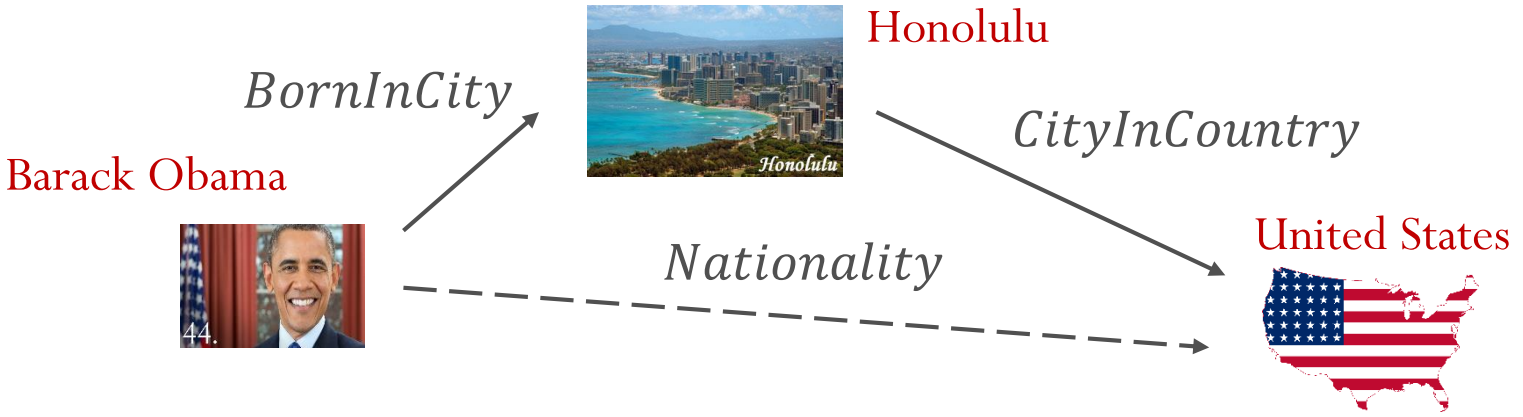
Empirical Comparisons of NN-based KB Embedding Methods [Yang+ ICLR-2015]

- Models with fewer parameters tend to perform better (for the datasets FB-15k and WN).
- The bilinear operator ($a^T M_r b$) plays an important role in capturing entity interactions.
- With the same model complexity, multiplicative operations are superior to additive operations in modeling relations.
- Initializing entity vectors with pre-trained phrase embedding vectors can significantly boost performance.

Mining Horn-clause Rules [Yang+ ICLR-2015]

- Can relation embedding capture relation composition?

$$BornInCity(a,b) \wedge CityInCountry(b,c) \Rightarrow Nationality(a,c)$$



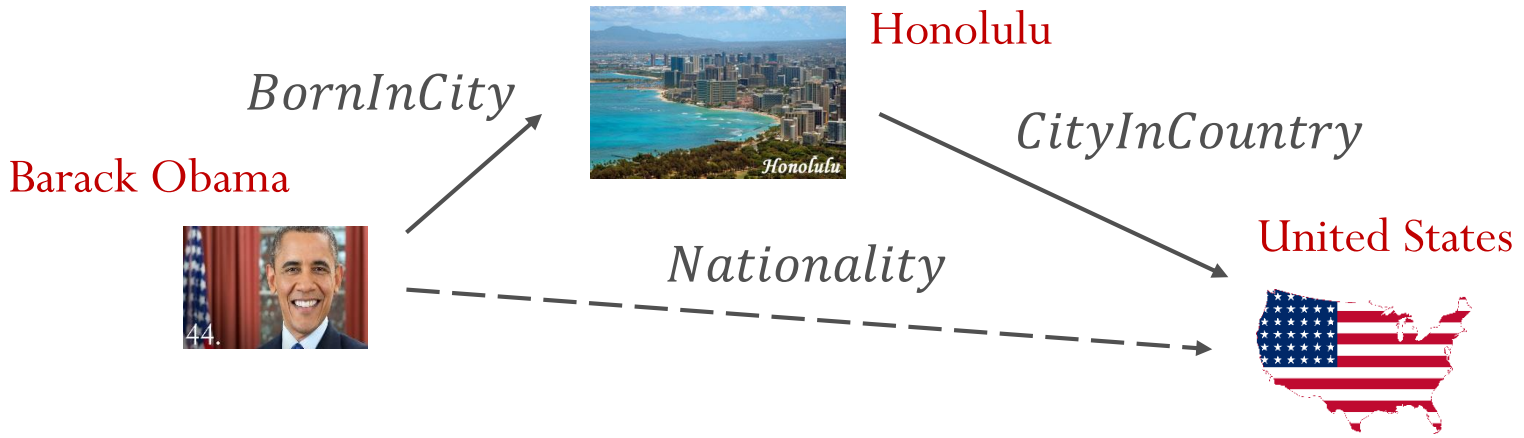
- Embedding-based Horn-clause rule extraction
  - For each relation r, find a chain of relations $r_1 \cdots r_n$, such that:

$$dist(M_r, M_1 \circ M_2 \circ \cdots \circ M_n) < \theta$$

  - $r_1(e_1,e_2) \wedge r_2(e_2,e_3) \cdots \wedge r_n(e_n,e_{n+1}) \rightarrow r(e_1,e_{n+1})$

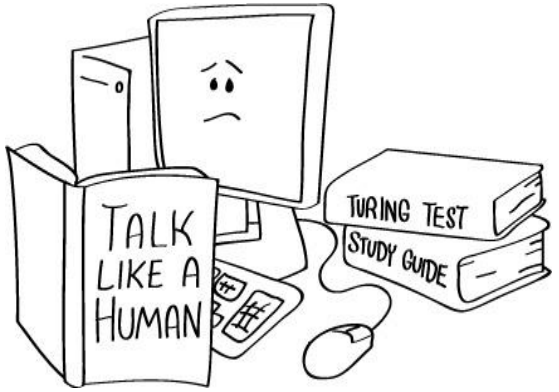
Learning from Relational Paths [Guu+ EMNLP-15, Garcia-Duran+ EMNLP-15, Toutanova+ ACL-16]

- Single-edge path: $\mathbf{score}(s, r, t) = v_s^{\mathrm{T}} M_r v_t$
  - (Obama, Nationality, USA)
- Multi-edge path: $\mathbf{score}(s, r_1, \cdots, r_k, t) = v_s^{\mathrm{T}} M_{r_1} \cdots M_{r_k} v_t$
  - (Obama, BornInCity, CityInCountry, USA)



Natural Language Understanding

- Continuous Word Representations & Lexical Semantics
- Knowledge Base Embedding
- KB-based Question Answering & Machine Comprehension
  - Yih & Ma, "Question Answering with Knowledge Bases, Web and Beyond." Tutorial in NAACL-HLT-16, SIGIR-16 http://aka.ms/tutorialQA



http://csunplugged.org/turing-test

Who is Justin Bieber's sister?

Jazmyn Bieber

semantic parsing

$\lambda x.\, \text{sister\_of}(\text{justin\_bieber}, x)$

matching

$\text{sibling\_of}(\text{justin\_bieber}, x) \wedge \text{gender}(x, \text{female})$

query

Knowledge Base

WebQuestions Dataset [Berant+ EMNLP-2013]

- *What character did Natalie Portman play in Star Wars?* ⇒ Padme Amidala
- *What kind of money to take to Bahamas?* ⇒ Bahamian dollar
- *What currency do you use in Costa Rica?* ⇒ Costa Rican colon
- *What did Obama study in school?* ⇒ political science
- *What do Michelle Obama do for a living?* ⇒ writer, lawyer
- *What killed Sammy Davis Jr?* ⇒ throat cancer

[Examples from Berant]

- 5,810 questions crawled from Google Suggest API and answered using Amazon MTurk
  - 3,778 training, 2,032 testing
  - A question may have multiple answers → using Avg. F1 (~accuracy)

Key Challenge – Language Mismatch

- Lots of ways to ask the same question
  - *"What was the date that Minnesota became a state?"*
  - *"Minnesota became a state on?"*
  - *"When was the state Minnesota created?"*
  - *"Minnesota's date it entered the union?"*
  - *"When was Minnesota established as a state?"*
  - *"What day did Minnesota officially become a state?"*

- Need to map them to the predicate defined in KB
  - location.dated_location.date_founded

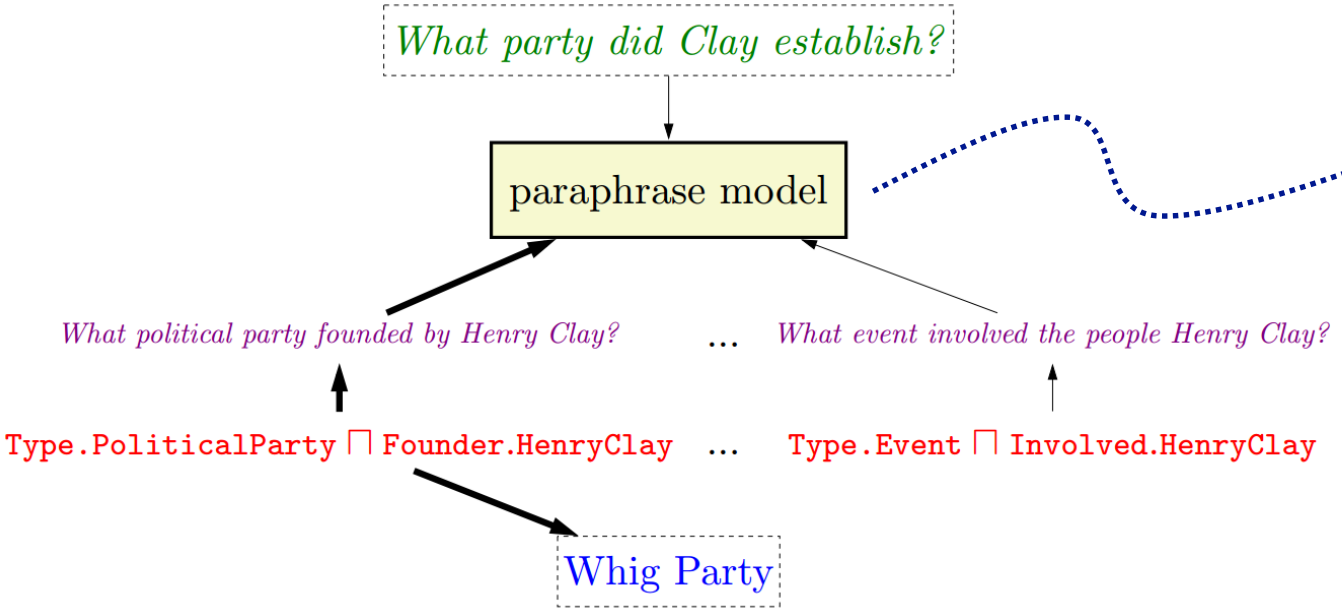
Matching Question and Relation

- Similar text can be mapped to very different relations
  - Q = *Who is the father of King George VI?*
  - R = people.person.parents
  - Q = *Who is the father of the Periodic Table?*
  - R = law.invention.inventor

- Estimate $P(R|Q)$ using naïve Bayes [Yao&VanDurme ACL-14]
  - $P(R|Q) \propto P(Q|R)P(R) \approx \prod_w P(w|R)P(R)$
  - Use ClueWeb09 dataset with Freebase entity annotations to create a "relation – sentence" parallel corpus
  - Derive $P(w|R)$ and $P(R)$ from the word alignment model (IBM Model 1)
  - Top words for film.film.directed_by: won, start, among, show.

Matching Questions

- Semantic Parsing via Paraphrasing [Berant&Liang ACL-14]



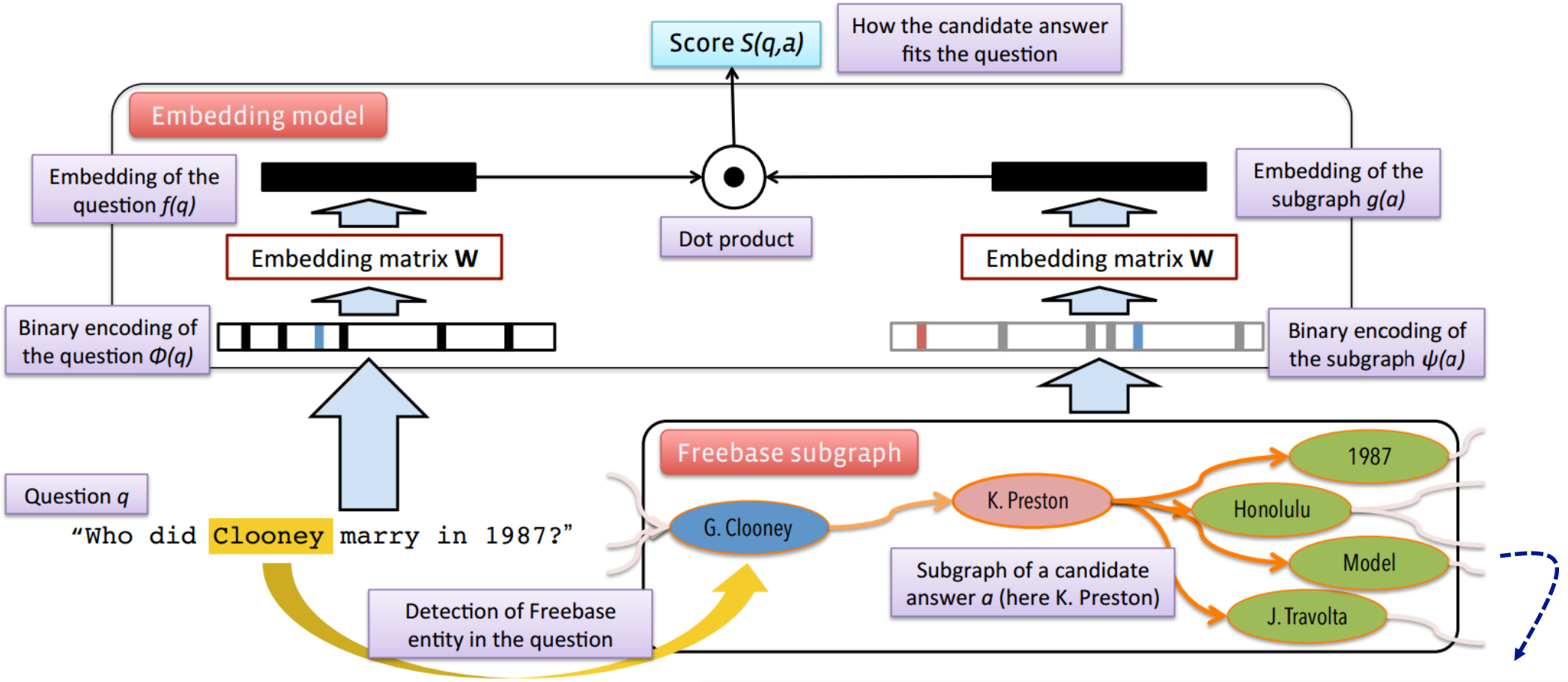
Learned using question paraphrases in Paralex [Fader+ ACL-13]

- Create phrase matching features using phrase table derived from word alignment results
- Represent questions as vectors (avg. of word vectors)

Subgraph Embedding [Bordes+ EMNLP-2014]

- Basic idea: map question and answer to vectors
  - q: question (Who did Clooney marry in 1987?)
  - a: answer candidate (K. Preston)
  - $S(q,a) = f(q)^{\mathrm{T}} g(a)$, where $f(q) = \mathbf{W}\phi(q), g(a) = \mathbf{W}\psi(a)$
- Answer candidate generation
  - Assume the topic entity (Clooney → G. Clooney) in q is given
  - All neighboring entities 1 or 2 edges away from topic entity
- Input encoding
  - $\phi(q)$: bag-of-word binary vectors
  - $\psi(a)$: binary encoding of the answer entity

Subgraph Embedding [Bordes+ EMNLP-2014]

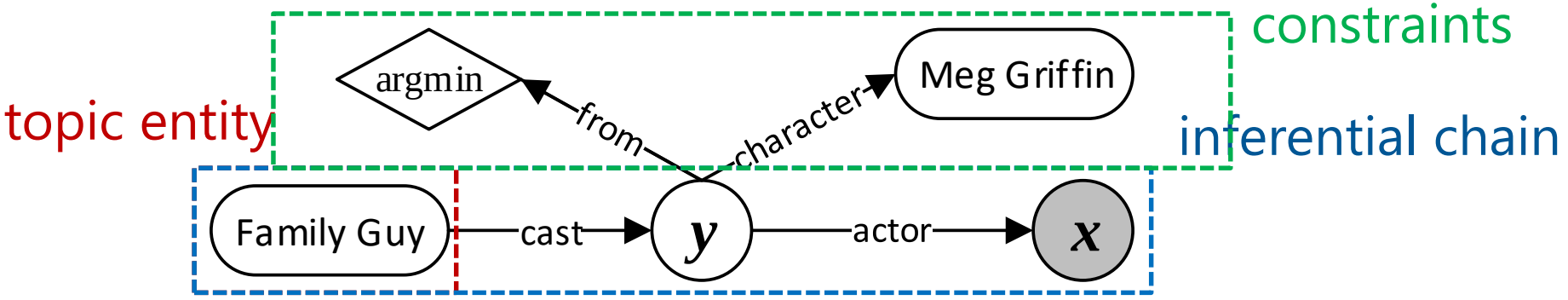


Other candidate answer encoding that includes the path, or other neighboring entities (subgraph)

Staged Query Graph Generation [Yih+ ACL-15]

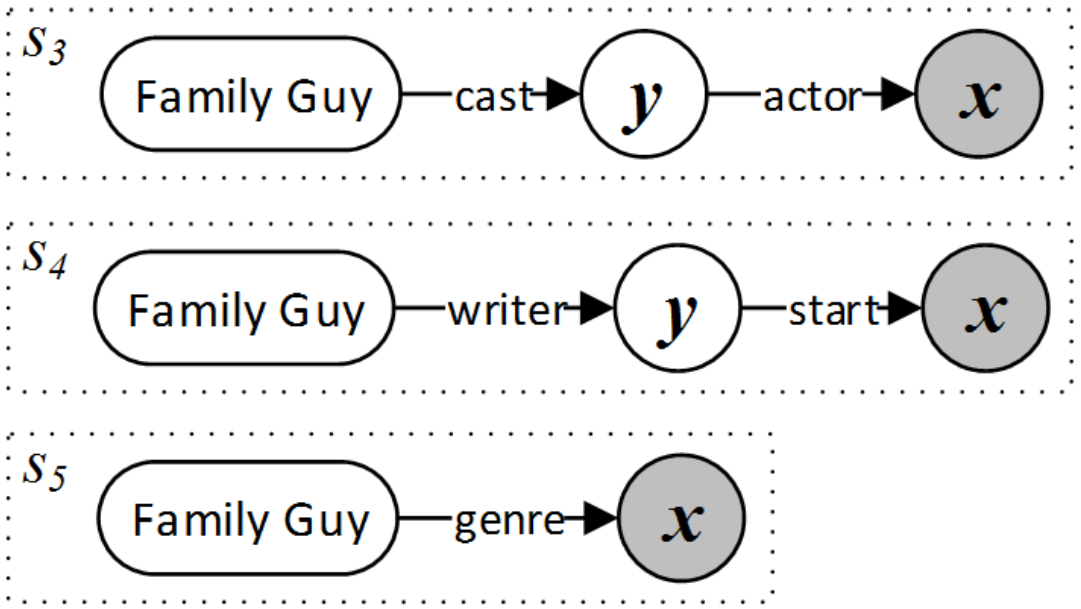
- Query graph
  - Resembles subgraphs of the knowledge base
  - Can be directly mapped to a logical form in λ-calculus
  - Semantic parsing: a search problem that *grows* the graph through actions

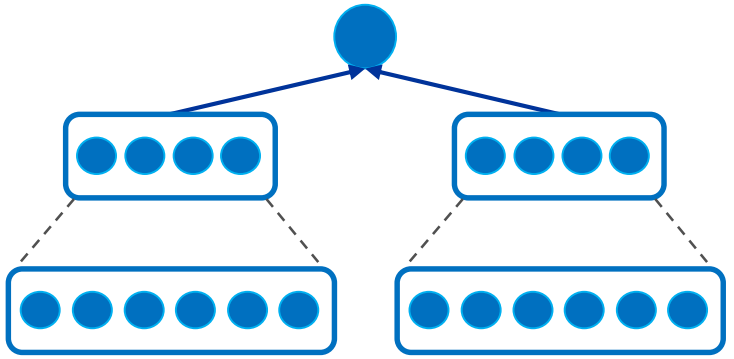
- Who first voiced Meg on Family Guy?
- $\lambda x.\exists y.\mathrm{cast}(\mathrm{FamilyGuy}, y) \wedge \mathrm{actor}(y, x) \wedge \mathrm{character}(y, \mathrm{MegGriffin})$



Identify Inferential Chain using DSSM

- Who first voiced Meg on Family Guy?





- Semantic match (“Who first voiced Meg on ⟨e⟩”, “cast-actor”)
- Single pattern/relation matching model: 49.6% F_1 (vs. 52.5% F_1 Full)

DeepMind Q&A Dataset [Hermann et al., NIPS-15]

- High-level dataset creation process
  - Pick a large corpus (e.g., news articles, stories)
  - Develop an (almost) automatic way to generate (fill-in-the-blank) questions

- 93k CNN & 220k Daily Mail articles
- Bullet points (summary / paraphrases) → Cloze questions
  - Replacing one entity with a placeholder
  - ~4 questions per document
  - ~1M document / query / answer triples

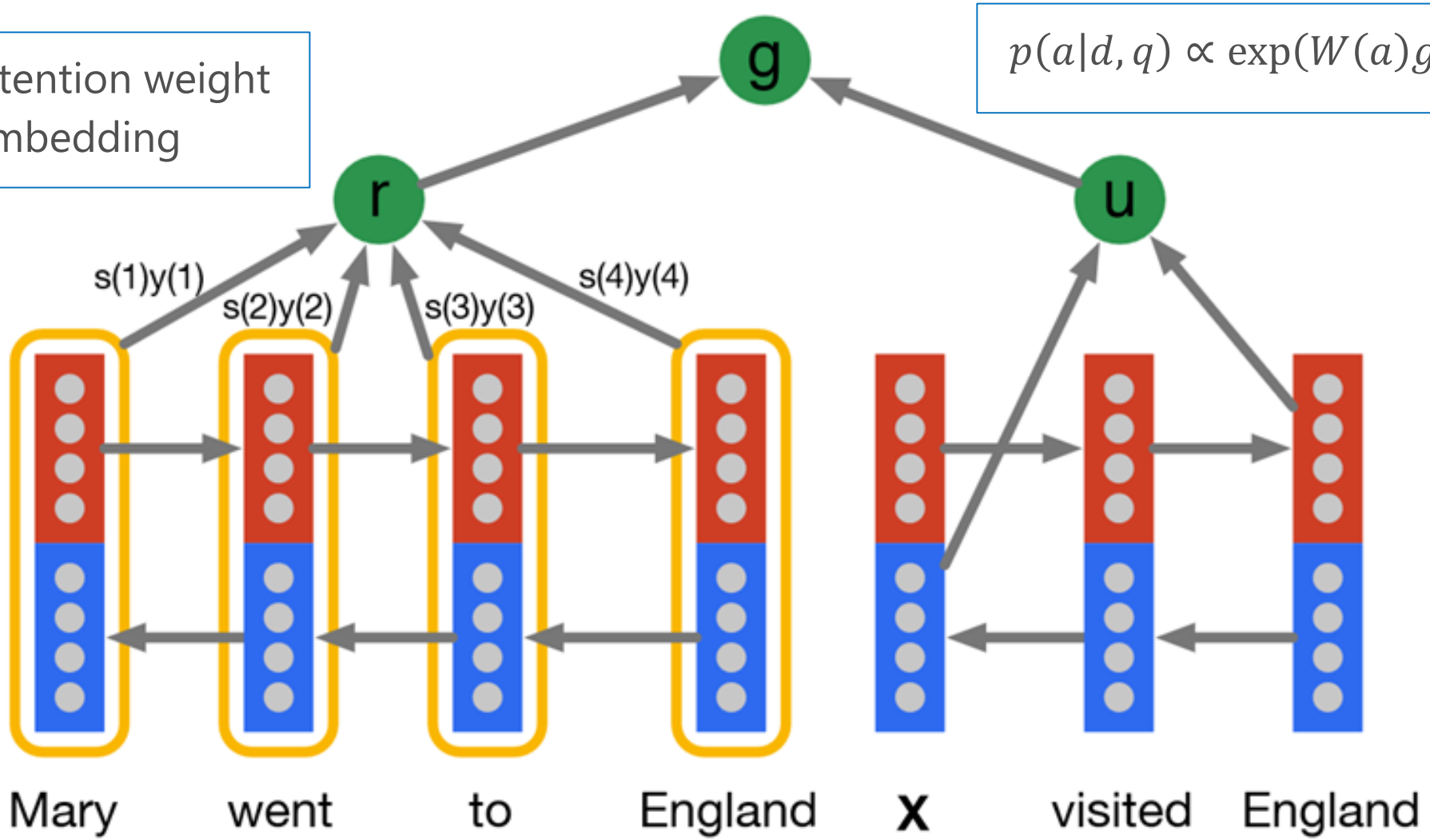
Example [Hermann et al., NIPS-15. Table 3]

Original Version	Anonymised Version
Context	
The BBC producer allegedly struck by Jeremy Clarkson will not press charges against the "Top Gear" host, his lawyer said Friday. Clarkson, who hosted one of the most-watched television shows in the world, was dropped by the BBC Wednesday after an internal investigation by the British broadcaster found he had subjected producer Oisin Tymon "to an unprovoked physical and verbal attack." ...	the *ent381* producer allegedly struck by *ent212* will not press charges against the " *ent153* " host , his lawyer said friday . *ent212* , who hosted one of the most - watched television shows in the world , was dropped by the *ent381* wednesday after an internal investigation by the *ent180* broadcaster found he had subjected producer *ent193* " to an unprovoked physical and verbal attack . " ...
Query	
Producer **X** will not press charges against Jeremy Clarkson, his lawyer says.	producer **X** will not press charges against *ent212* , his lawyer says .
Answer	
Oisin Tymon	*ent193*

Neural Network Models – Attentive Reader

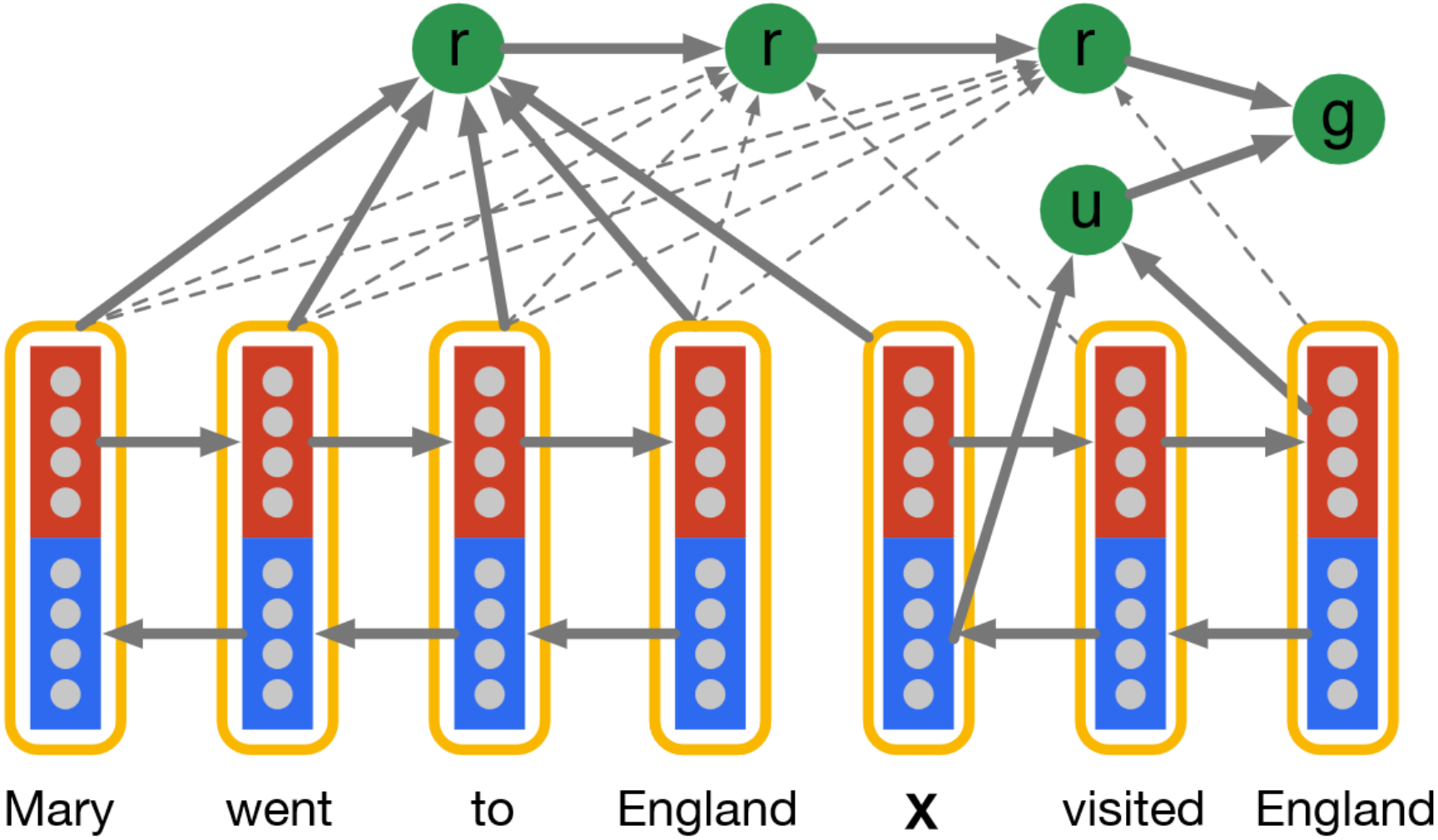
s: attention weight
y: embedding

$$p(a|d,q) \propto \exp(W(a)g(d,q))$$



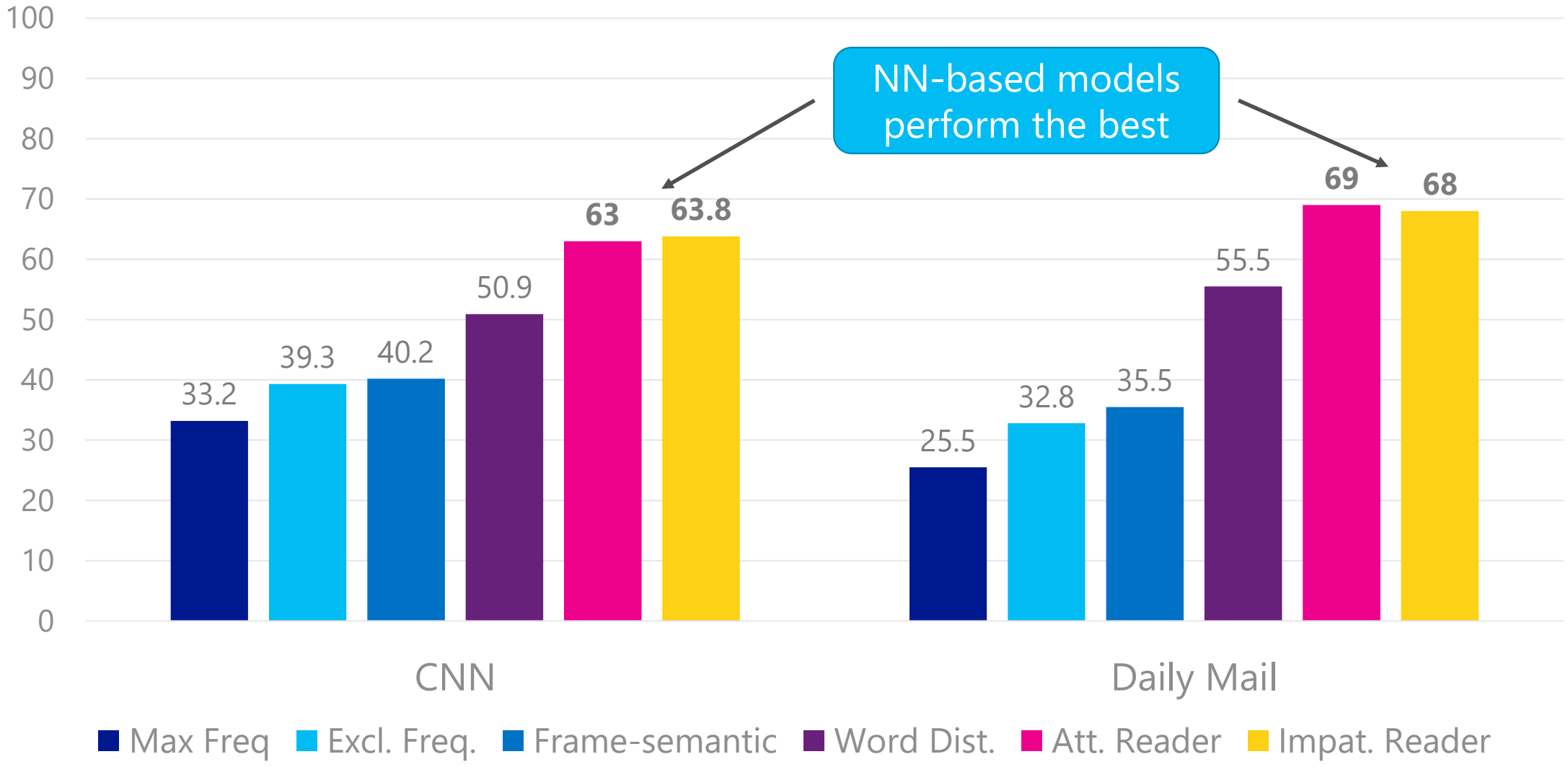
[Hermann et al., NIPS-15. Fig 1a]

Neural Network Models – Impatient Reader



[Hermann et al., NIPS-15. Fig 1b]

Accuracy


100
90
80
70
60
50
40
30
20
10
0
NN-based models perform the best
33.2
39.3
40.2
50.9
63
63.8
25.5
32.8
35.5
55.5
69
68
CNN
Daily Mail
Max Freq
Excl. Freq.
Frame-semantic
Word Dist.
Att. Reader
Impat. Reader

A Thorough Examination… [Chen et al. ACL-16]

- Challenges & Questions
  - A clever way of creating large supervised data, but an artificial task
  - Unclear what level of reading comprehension needed
- Good News – The task is not really difficult!
  - An entity-centric classifier with simple features works comparably
  - A variant of the Attentive Reader model achieves the new best result
- Bad News – The task is not really difficult!
  - Not much "comprehension" is needed
  - Probably reached the ceiling (25% questions unanswerable)

Interim summary

Continuous-space representations are effective for several natural language semantic tasks

- Continuous Word Representations & Lexical Semantics
- Knowledge Base Embedding
- KB-based Question Answering & Machine Comprehension

Data & tools (partial list)

- Word2Vec https://code.google.com/p/word2vec/
- GloVe http://nlp.stanford.edu/projects/glove/
- MSR Continuous Space Text Representation http://aka.ms/msrcstr
- DeepMind Q&A dataset http://cs.nyu.edu/~kcho/DMQA/

Conclusions

- Exciting advances in NN and continuous representations
  - **Text processing & Knowledge reasoning**

- Looking forward
  - **Building an universal intelligence space**
    - Text, Knowledge, Reasoning, …
    - Sent2Vec (DSSM) http://aka.ms/sent2vec
  - **From component models to end-to-end solutions**

References

- Andreas, J., Rohrbach, M., Darrell, T., Klein, D., 2016. Neural Module Networks, CVPR
- Auli, M., Galley, M., Quirk, C. and Zweig, G., 2013. Joint language and translation modeling with recurrent neural networks. In EMNLP.
- Auli, M., and Gao, J., 2014. Decoder integration and expected bleu training for recurrent neural network language models. In ACL.
- Bahdanau, D., Cho, K., and Bengio, Y. 2015. Neural machine translation by joingly learning to align and translate, in ICLR 2015.
- Bejar, I., Chaffin, R. and Embretson, S. 1991. Cognitive and psychometric analysis of analogical problem solving. Recent research in psychology.
- Bengio, Y., 2009. Learning deep architectures for AI. Foundumental Trends Machine Learning, vol. 2.
- Bengio, Y., Courville, A., and Vincent, P. 2013. Representation learning: A review and new perspectives. IEEE Trans. PAMI, vol. 38, pp. 1798-1828.
- Bengio, Y., Ducharme, R., and Vincent, P., 2000. A Neural Probabilistic Language Model, in NIPS.
- Berant, J., Chou, A., Frostig, R., Liang, P. 2013. Semantic Parsing on Freebase from Question-Answer Pairs. In EMNLP.
- Berant, J., and Liang, P. 2014. Semantic parsing via paraphrasing. In ACL.
- Bian, J., Gao, B., Liu, T. 2014. Knowledge-Powered Deep Learning for Word Embedding. In ECML.
- Blei, D., Ng, A., and Jordan M. 2001. Latent dirichlet allocation. In NIPS.
- Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J. and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In NIPS.
- Bordes, A., Chopra, S., and Weston, J. 2014. Question answering with subgraph embeddings. In EMNLP.
- Bordes, A., Glorot, X., Weston, J. and Bengio Y. 2012. Joint Learning of Words and Meaning Representations for Open-Text Semantic Parsing. In AISTATS.
- Brown, P., deSouza, P. Mercer, R., Della Pietra, V., and Lai, J. 1992. Class-based n-gram models of natural language. Computational Linguistics 18 (4).
- Chandar, A. P. S., Lauly, S., Larochelle, H., Khapra, M. M., Ravindran, B., Raykar, V., and Saha, A. (2014). An autoencoder approach to learning bilingual word representations. In NIPS.
- Chang, K., Yih, W., and Meek, C. 2013. Multi-Relational Latent Semantic Analysis. In EMNLP.
- Chang, K., Yih, W., Yang, B., and Meek, C. 2014. Typed Tensor Decomposition of Knowledge Bases for Relation Extraction. In EMNLP.
- Chen, D., Bolton, J., and Manning, C. 2016. A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task. In ACL.
- Collobert, R., and Weston, J. 2008. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In ICML.
- Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P., 2011. Natural language processing (almost) from scratch. in JMLR, vol. 12.
- Cui, L., Zhang, D., Liu, S., Chen, Q., Li, M., Zhou, M., and Yang, M. (2014). Learning topic representation for SMT with neural networks. In ACL.
- Dahl, G., Yu, D., Deng, L., and Acero, 2012. A. Context-dependent, pre-trained deep neural networks for large vocabulary speech recognition, IEEE Trans. Audio, Speech, & Language Proc., Vol. 20 (1), pp. 30-42.
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T., and Harshman, R. 1990. Indexing by latent semantic analysis. J. American Society for Information Science, 41(6): 391-407

References

- Deng, L., He, X., Gao, J., 2013. Deep stacking networks for information retrieval, ICASSP
- Deng, L., Seltzer, M., Yu, D., Acero, A., Mohamed, A., and Hinton, G., 2010. Binary Coding of Speech Spectrograms Using a Deep Auto-encoder, in Interspeech.
- Deng, L., Tur, G, He, X, and Hakkani-Tur, D. 2012. Use of kernel deep convex networks and end-to-end learning for spoken language understanding, Proc. IEEE Workshop on Spoken Language Technologies.
- Deng, L., Yu, D. and Acero, A. 2006. Structured speech modeling, IEEE Trans. on Audio, Speech and Language Processing, vol. 14, no. 5, pp. 1492-1504.
- Deng, L., Yu, D., and Platt, J. 2012. Scalable stacking and learning for building deep architectures, Proc. ICASSP.
- Deng, L. and Yu, D. 2014. Deeping learning methods and applications. Foundations and Trends in Signal Processing 7:3-4.
- Deoras, A., and Sarikaya, R., 2013. Deep belief network based semantic taggers for spoken language understanding, in INTERSPEECH.
- Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., and Makhoul, J., 2014. Fast and Robust Neural Network Joint Models for Statistical Machine Translation, ACL.
- Devlin, J., Cheng, H., Fang, H., Gupta, S., Deng, L., He, X., Zweig, G., and Mitchell, M., 2015. Language Models for Image Captioning: The Quirks and What Works, ACL
- Duh, K. 2014. Deep learning for natural language processing and machine translation. Tutorial. 2014.
- Duh, K., Neubig, G., Sudoh, K., and Tsukada, H. (2013). Adaptation data selection using neural language models: Experiments in machine translation. In ACL.
- Fader, A., Zettlemoyer, L., and Etzioni, O. 2013. Paraphrase-driven learning for open question answering. In ACL.
- Fang, H., Gupta, S., Iandola, F., Srivastava, R., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J., Zitnick, L., Zweig, G., "From Captions to Visual Concepts and Back," arXiv:1411.4952
- Faruqui, M. and Dyer, C. (2014). Improving vector space word representations using multilingual correlation. In EACL.
- Faruqui, M., Dodge, J., Jauhar, S., Dyer, C., Hovy, E., Smith, N. 2015. Retrofitting Word Vectors to Semantic Lexicons. In NAACL-HLT.
- Faruqui, M., Tsvetkov, Y., Yogatama, D., Dyer, C., Smith, N. 2015. Sparse Overcomplete Word Vector Representations. In ACL.
- Firth, J. R. 1957. *Papers in Linguistics 1934–1951*, Oxford University Press, 1957
- Frome, A., Corrado, G., Shlens, J., Bengio, S., Dean, J., Ranzato, M., and Mikolov, T., 2013. DeViSE: A Deep Visual-Semantic Embedding Model, Proc. NIPS.
- Galárraga, L., Teflioudi, C., Hose, K., Suchanek, F. 2013. Association Rule Mining Under Incomplete Evidence in Ontological Knowledge Bases. In WWW.
- Gao, J., He, X., Yih, W-t., and Deng, L. 2014a. Learning continuous phrase representations for translation modeling. In ACL.
- Gao, J., He, X., and Nie, J-Y. 2010. Clickthrough-based translation models for web search: from word models to phrase models. In CIKM.

References

- Gao, J., Pantel, P., Gamon, M., He, X., Deng, L., and Shen, Y. 2014b. Modeling interestingness with deep neural networks. In EMNLP
- Gao, J., Toutanova, K., Yih., W-T. 2011. Clickthrough-based latent semantic models for web search. In SIGIR.
- Gao, J., Yuan, W., Li, X., Deng, K., and Nie, J-Y. 2009. Smoothing clickthrough data for web search ranking. In SIGIR.
- Gao, J., and He, X. 2013. Training MRF-based translation models using gradient ascent. In NAACL-HLT.
- Getoor, L., and Taskar, B. editors. 2007. Introduction to Statistical Relational Learning. The MIT Press.
- Graves, A., Jaitly, N., and Mohamed, A., 2013a. Hybrid speech recognition with deep bidirectional LSTM, Proc. ASRU.
- Graves, A., Mohamed, A., and Hinton, G., 2013. Speech recognition with deep recurrent neural networks, Proc. ICASSP.
- Guu, K., Miller, J., & Liang, P. (2015). Traversing knowledge graphs in vector space. EMNLP-2015
- He, J., Chen, J., He, X., Gao, J., Li, L., Deng, L., Ostendorf, M., 2015 Deep Reinforcement Learning with an Action Space Defined by Natural Language, arXiv:1511.04636
- He, X. and Deng, L., 2013. Speech-Centric Information Processing: An Optimization-Oriented Approach, in Proceedings of the IEEE.
- He, X. and Deng, L., 2012. Maximum Expected BLEU Training of Phrase and Lexicon Translation Models , ACL.
- He, X., Deng, L., and Chou, W., 2008. Discriminative learning in sequential pattern recognition, Sept. IEEE Sig. Proc. Mag.
- Hermann, K. M. and Blunsom, P. (2014). Multilingual models for compositional distributed semantics. In ACL.
- Hermann, K. M., Kocisky, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In NIPS.
- Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., and Kingsbury, B., 2012. Deep Neural Networks for Acoustic Modeling in Speech Recognition, IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82-97.
- Hinton, G., Osindero, S., and The, Y-W. 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18: 1527-1554.
- Hinton, G., and Salakhutdinov, R., 2010. Discovering binary codes for documents by learning deep generative models. Topics in Cognitive Science.
- Hu, Y., Auli, M., Gao, Q., and Gao, J. 2014. Minimum translation modeling with recurrent neural networks. In EACL.
- Huang, E., Socher, R., Manning, C, and Ng, A. 2012. Improving word representations via global context and multiple word prototypes, Proc. ACL.
- Huang, P., He, X., Gao, J., Deng, L., Acero, A., and Heck, L. 2013. Learning deep structured semantic models for web search using clickthrough data. In CIKM.
- Hutchinson, B., Deng, L., and Yu, D., 2012. A deep architecture with bilinear modeling of hidden representations: Applications to phonetic recognition, Proc. ICASSP.
- Hutchinson, B., Deng, L., and Yu, D., 2013. Tensor deep stacking networks, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 35, pp. 1944 - 1957.
- Jansen, P., Surdeanu, M., Clark, P. 2014. Discourse Complements Lexical Semantics for Non-factoid Answer Reranking. In ACL.

References

- Jurgens, D., Mohammad, S., Turney, P. and Holyoak, K. 2012. SemEval-2012 Task 2: Measuring degrees of relational similarity. In SemEval.
- Kafle, K., Kanan, C., 2016. Answer-Type Prediction for Visual Question Answering, CVPR
- Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models., in EMNLLP
- Kiros, R., Zemel, R., and Salakhutdinov, R. 2013. Multimodal Neural Language Models, Proc. NIPS Deep Learning Workshop.
- Klementiev, A., Titov, I., and Bhattarai, B. (2012). Inducing crosslingual distributed representations of words. In COLING.
- Kocisky, T., Hermann, K. M., and Blunsom, P. (2014). Learning bilingual word representations by marginalizing alignments. In ACL.
- Koehn, P. 2009. Statistical Machine Translation. Cambridge University Press.
- Krizhevsky, A., Sutskever, I, and Hinton, G., 2012. ImageNet Classification with Deep Convolutional Neural Networks, NIPS.
- Landauer. T., 2002. On the computational basis of learning and cognition: Arguments from LSA. Psychology of Learning and Motivation, 41:43–84.
- Lao, N., Mitchell, T., and Cohen, W. 2011. Random walk inference and learning in a large scale knowledge base. In EMNLP.
- Lauly, S., Boulanger, A., and Larochelle, H. (2013). Learning multilingual word representations using a bag-of-words autoencoder. In NIPS.
- Le, H-S, Oparin, I., Allauzen, A., Gauvain, J-L., Yvon, F., 2013. Structured output layer neural network language models for speech recognition, IEEE Transactions on Audio, Speech and Language Processing.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. 1998. Gradient-based learning applied to document recognition, Proceedings of the IEEE, Vol. 86, pp. 2278-2324.
- Levy, O., and Goldberg, Y. 2014. Linguistic Regularities in Sparse and Explicit Word Representations. In CoNLL.
- Levy, O., and Goldberg, Y. 2014. Neural Word Embeddings as Implicit Matrix Factorization. In NIPS.
- Li, P., Hastie, T., and Church, K.. 2006. Very sparse random projections, in Proc. SIGKDD.
- Li, P., Liu, Y., and Sun, M. (2013). Recursive autoencoders for ITG-based translation. In EMNLP.
- Li, P., Liu, Y., Sun, M., Izuha, T., and Zhang, D. (2014b). A neural reordering model for phrase-based translation. In COLING.
- Liu, S., Yang, N., Li, M., and Zhou, M. (2014). A recursive recurrent neural network for statistical machine translation. In ACL.
- Liu, L., Watanabe, T., Sumita, E., and Zhao, T. (2013). Additive neural networks for statistical machine translation. In ACL.
- Liu, X., Gao, J., He, X., Deng, L., Duh, K., Wang, Y., 2015. Representation learning using multi-task deep neural networks for semantic classification and information retrieval, NAACL
- Lu, S., Chen, Z., and Xu, B. (2014). Learning new semi-supervised deep auto-encoder features for statistical machine translation. In ACL.
- Maskey, S., and Zhou, B. 2012. Unsupervised deep belief feature for speech translation, in ICASSP.
- Mesnil, G., He, X., Deng, L., and Bengio, Y., 2013. Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding, in Interspeech.

References

- Mikolov, T., Kombrink, S., Burget, L., Cernocky, J., Khudanpur, S. 2011. Extensions of recurrent neural network based language model. In ICASSP.
- Mikolov, T. 2012. Statistical Language Models based on Neural Networks, Ph.D. thesis, Brno University of Technology.
- Mikolov, T., Chen, K., Corrado, G., and Dean, J. 2013. Efficient estimation of word representations in vector space, Proc. ICLR.
- Mikolov, T., Kombrink,. S., Burget, L., Cernocky, J., Khudanpur, S., 2011. Extensions of Recurrent Neural Network LM. ICASSP.
- Mikolov, T., Yih, W., Zweig, G., 2013. Linguistic Regularities in Continuous Space Word Representations. In NAACL-HLT.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. 2013. Distributed Representations of Words and Phrases and their Compositionality. In NIPS.
- Mnih, A., Kavukcuoglu, K. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In NIPS.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing Atari with Deep Reinforcement Learning, NIPS
- Mohamed, A., Yu, D., and Deng, L. 2010. Investigation of full-sequence training of deep belief networks for speech recognition, Proc. Interspeech.
- Mohammad, S., Dorr, Bonnie., and Hirst, G. 2008. Computing word pair antonymy. In EMNLP.
- Narasimhan, K., Kulkarni, T., Barzilay, R., 2015. Language Understanding for Text-based Games Using Deep Reinforcement Learning. EMNLP
- Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., and Ng, A. 2011. Multimodal deep learning, Proc. ICML.
- Nickel, M., Tresp, V., and Kriegel, H. 2011. A three-way model for collective learning on multi-relational data. In ICML.
- Niehues, J., Waibel, A. 2013. Continuous space language models using Restricted Boltzmann Machines. In IWLT.
- Noh, H., Seo, P., Han, B., 2016. Image Question Answering Using Convolutional Neural Network With Dynamic Parameter Prediction, CVPR
- Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., Song, X., Ward R., 2016. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval, IEEE/ACM Transactions on Audio, Speech, and Language Processing 24 (4), 694-707
- Pennington, J., Socher, R., Manning, C. 2014. Glove: Global Vectors for Word Representation. In EMNLP.
- Reddy, S., Lapata, M., and Steedman, M. 2014. Large-scale semantic parsing without question-answer pairs. Transactions of the Association for Computational Linguistics (TACL).
- Sainath, T., Mohamed, A., Kingsbury, B., and Ramabhadran, B. 2013. Convolutional neural networks for LVCSR, Proc. ICASSP.
- Salakhutdinov R., and Hinton, G., 2007 Semantic hashing. in Proc. SIGIR Workshop Information Retrieval and Applications of Graphical Models
- Salton, G. and McGill, M. 1983. Introduction to Modern Information Retrieval. McGraw Hill.
- Sarikaya, R., Hinton, G., and Ramabhadran, B., 2011. Deep belief nets for natural language call-routing, in Proceedings of the ICASSP.
- Schwenk, H. 2012. Continuous space translation models for phrase-based statistical machine translation, in COLING.
- Schwenk, H., Rousseau, A., and Attik, M., 2012. Large, pruned or continuous space language models on a gpu for statistical machine translation, in NAACL-HLT 2012 Workshop.

References

- Seide, F., Li, G., and Yu, D. 2011. Conversational speech transcription using context-dependent deep neural networks, Proc. Interspeech
- Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. 2014. Learning Semantic Representations Using Convolutional Neural Networks for Web Search, in Proceedings of WWW.
- Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. 2014. A convolutional latent semantic model for web search. CIKM
- Shih, K., Singh, S., Hoiem, D., 2016. Where to Look: Focus Regions for Visual Question Answering, CVPR
- Silver, D., Huang, A., Maddison, C., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., 2016. Mastering the game of Go with deep neural networks and tree search, Nature
- Simonyan, K., Zisserman, A., 2015 Very Deep Convolutional Networks for Large-Scale Image Recognition. ICLR 2015
- Socher, R., Chen, D., Manning, C., and Ng, A. 2013. Reasoning With Neural Tensor Networks For Knowledge Base Completion. In NIPS.
- Socher, R., Huval, B., Manning, C., Ng, A., 2012. Semantic compositionality through recursive matrix-vector spaces. In EMNLP.
- Socher, R., Lin, C., Ng, A., and Manning, C. 2011. Learning continuous phrase representations and syntactic parsing with recursive neural networks, Proc. ICML.
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C., Ng A., and Potts. C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank, Proc. EMNLP
- Son, L. H., Allauzen, A., and Yvon, F. (2012). Continuous space translation models with neural networks. In NAACL.
- Song, X. He, X., Gao. J., and Deng, L. 2014. Unsupervised Learning of Word Semantic Embedding using the Deep Structured Semantic Model. MSR Tech Report.
- Song, Y., Wang, H., and He, X., 2014. Adapting Deep RankNet for Personalized Search. Proc. WSDM.
- Songyot, T. and Chiang, D. (2014). Improving word alignment using word similarity. In EMNLP.
- Sundermeyer, M., Alkhouli, T., Wuebker, J., and Ney, H. (2014). Translation modeling with bidirectional recurrent neural networks, in EMNLP.
- Sutton, R., Barto, A., 1998. Reinforcement Learning: An Introduction. MIT Press.
- Tamura, A., Watanabe, T., and Sumita, E. (2014). Recurrent neural networks for word alignment model. In ACL.
- Tapaswi, M., Zhu, Y., Stiefelhagen, R., Torralba, A., Urtasun, R., Fidler, S., 2016. MovieQA: Understanding Stories in Movies Through Question-Answering, CVPR
- Toutanova, K., Lin, X. V., Yih, W., Poon, H., & Quirk, C. (2016) Compositional Learning of Embeddings for Relation Paths in Knowledge Bases and Text. ACL-2016
- Tran, K. M., Bisazza, A., and Monz, C. (2014). Word translation prediction for morphologically rich languages with bilingual neural networks. In EMNLP.
- Tran, K., He, X., Zhang, L., Sun, J., Carapcea, C., Thrasher, C., Buehler, C., Sienkiewicz, C., “Rich Image Captioning in the Wild,” DeepVision, CVPR 2016

References

- Tur, G., Deng, L., Hakkani-Tur, D., and He, X., 2012. Towards Deeper Understanding Deep Convex Networks for Semantic Utterance Classification, in ICASSP.
- Turney P. 2008. A uniform approach to analogies, synonyms, antonyms, and associations. In COLING. Songyot, T. and Chiang, D. (2014). Improving word alignment using word similarity. In EMNLP.
- Vaswani, A., Zhao, Y., Fossum, V., and Chiang, D. 2013. Decoding with large-scale neural language models improves translation. In EMNLP.
- Wang, H., He, X., Chang, M., Song, Y., White, R., Chu, W., 2013. Personalized ranking model adaptation for web search, SIGIR
- Wang, Z., Zhang, J., Feng, J., Chen, Z. 2014. Knowledge Graph and Text Jointly Embedding. In EMNLP.
- Watkins, C., and Dayan, P., 1992. Q-learning. Machine Learning
- Wright, S., Kanevsky, D., Deng, L., He, X., Heigold, G., and Li, H., 2013. Optimization Algorithms and Applications for Speech and Language Processing, in IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 11.
- Wu, H., Dong, D., Hu, X., Yu, D., He, W., Wu, H., Wang, H., and Liu, T. (2014a). Improve statistical machine translation with context-sensitive bilingual semantic embedding model. In EMNLP.
- Wu, Q., Wang, P., Shen, C., Dick, A., Hengel, A., 2016. Ask Me Anything: Free-Form Visual Question Answering Based on Knowledge From External Sources, CVPR
- Wu, Y., Watanabe, T., and Hori, C. (2014b). Recurrent neural network-based tuple sequence model for machine translation. In COLING.
- Xu, C., Bai, Y., Bian, J., Gao, B., Wang, G., Liu, X., Liu, T. 2014. RC-NET: A General Framework for Incorporating Knowledge into Word Representations. In CIKM.
- Yang, B., Yih, W., He, X., Gao, J., and Deng L. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In ICLR.
- Yang, N., Liu, S., Li, M., Zhou, M., and Yu, N. 2013. Word alignment modeling with context dependent deep neural network. In ACL.
- Yang, Y., Chang, M. 2015. S-MART: Novel Tree-based Structured Learning Algorithms Applied to Tweet Entity Linking. In ACL.
- Yao, K., Zweig, G., Hwang, M-Y. , Shi, Y., Yu, D., 2013. Recurrent neural networks for language understanding, submitted to Interspeech.
- Yao, X., Van Durme, B. 2014. Information Extraction over Structured Data: Question Answering with Freebase. In ACL.
- Yann, D., Tur, G., Hakkani-Tur, D., Heck, L., 2014. Zero-Shot Learning and Clustering for Semantic Utterance Classification Using Deep Learning. In ICLR.
- Yogatama, D., Faruqui, M., Dyer, C., Smith, N. 2015. LearningWord Representations with Hierarchical Sparse Coding. In ICML.
- Yih, W., Toutanova, K., Platt, J., and Meek, C. 2011. Learning discriminative projections for text similarity measures. In CoNLL.
- Yih, W., Zweig, G., Platt, J. 2012. Polarity Inducing Latent Semantic Analysis. In EMNLP-CoNLL.
- Yih, W., Chang, M., Meek, C., Pastusiak, A. 2013. Question Answering Using Enhanced Lexical Semantic Models. In ACL.
- Yih, W., He, X., Meek, C. 2014. Semantic Parsing for Single-Relation Question Answering. In ACL.
- Yih, W., Chang, M., He, X., Gao, J. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base, In ACL.

References

- Yogatama, D., Faruqui, M., Dyer, C., & Smith, N. A. (2015, July). Learning word representations with hierarchical sparse coding. In Proc. of ICML.
- Zeiler, M. and Fergus, R. 2013. Visualizing and understanding convolutional networks, arXiv:1311.2901, pp. 1-11.
- Zhang, J., Liu, S., Li, M., Zhou, M., and Zong, C. (2014). Bilingually-constrained phrase embeddings for machine translation. In ACL.
- Zhu, Y., Groth, O., Bernstein, M., Fei-Fei, L., 2016. Visual7W: Grounded Question Answering in Images, CVPR
- Zou, W. Y., Socher, R., Cer, D., and Manning, C. D. (2013). Bilingual word embeddings for phrase-based machine translation. In EMNLP.