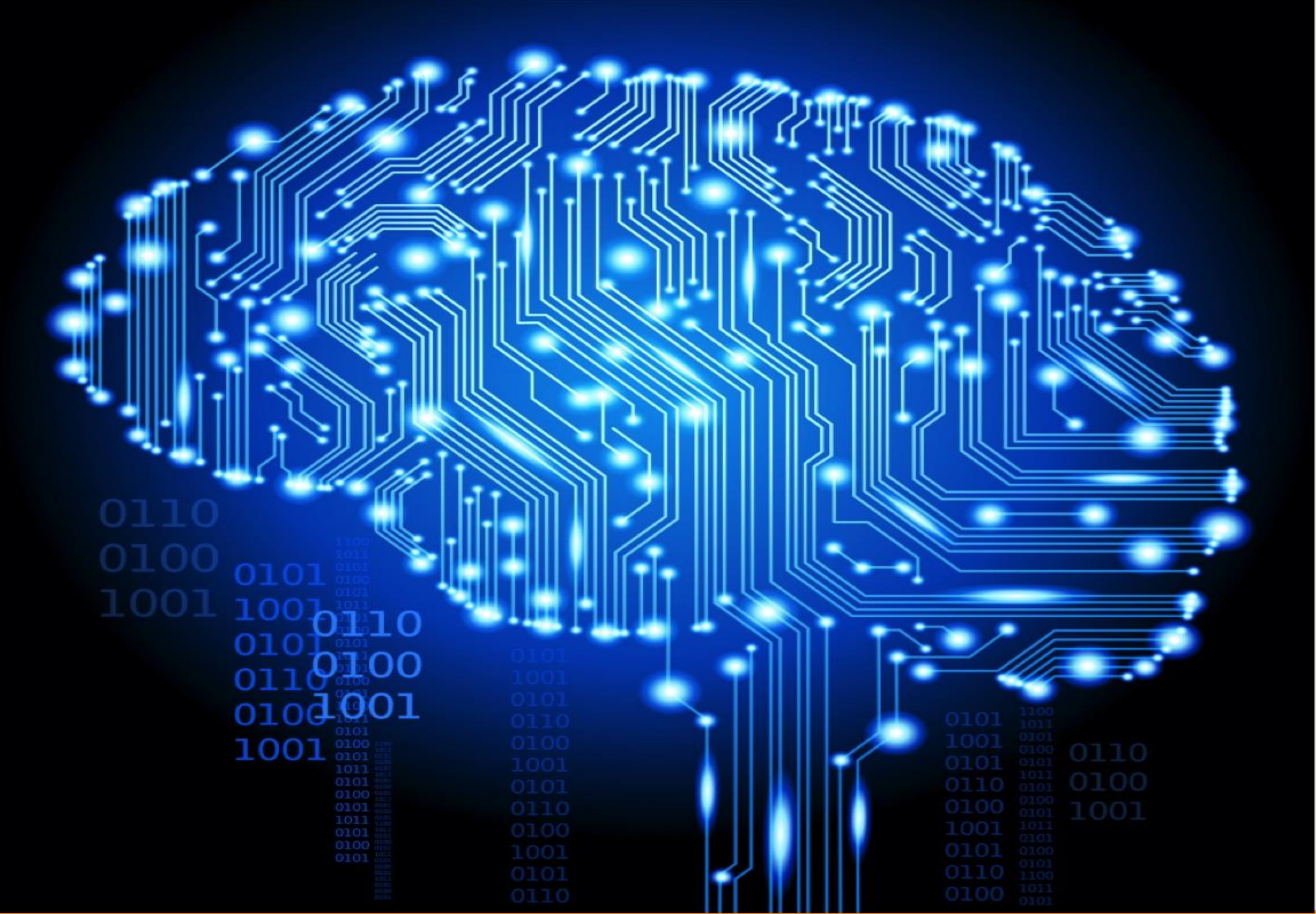


Lecture 10

Deep Learning@UvA

Previous Lecture

- Recurrent Neural Networks (RNN) for sequences
- Backpropagation Through Time
- RNNs using Long Short-Term Memory (LSTM)
- Applications of Recurrent Neural Networks

Lecture Overview

- Memory networks
- Recursive networks

Memory

Why memory? Example!

- *“Bilbo travelled to the cave. Gollum dropped the ring there. Bilbo took the ring. This sentence is random noise for illustration purposes. Bilbo went back to the Shire. Bilbo left the ring there. Frodo got the ring. Frodo journeyed to Mount-Doom. Frodo dropped the ring there. Sauron died. Frodo went back to the Shire. Bilbo travelled to the Grey-havens. The End.”*

Why memory? Example!

- *“Bilbo travelled to the cave. Gollum dropped the ring there. Bilbo took the ring. This sentence is random noise for illustration purposes. Bilbo went back to the Shire. Bilbo left the ring there. Frodo got the ring. Frodo journeyed to Mount-Doom. Frodo dropped the ring there. Sauron died. Frodo went back to the Shire. Bilbo travelled to the Grey-havens. The End.”*
- *“Q: Where is the ring?”* → *“A: Mount-Doom”*

Why memory? Example!

- *“Bilbo travelled to the cave. Gollum dropped the ring there. Bilbo took the ring. This sentence is random noise for illustration purposes. Bilbo went back to the Shire. Bilbo left the ring there. Frodo got the ring. Frodo journeyed to Mount-Doom. Frodo dropped the ring there. Sauron died. Frodo went back to the Shire. Bilbo travelled to the Grey-havens. The End.”*
- *“Q: Where is the ring?”* → *“A: Mount-Doom”*
- *“Q: Where is Bilbo now?”* → *“A: Grey-havens”*

Why memory? Example!

- *“Bilbo travelled to the cave. Gollum dropped the ring there. Bilbo took the ring. This sentence is random noise for illustration purposes. Bilbo went back to the Shire. Bilbo left the ring there. Frodo got the ring. Frodo journeyed to Mount-Doom. Frodo dropped the ring there. Sauron died. Frodo went back to the Shire. Bilbo travelled to the Grey-havens. The End.”*
- *“Q: Where is the ring?”* → *“A: Mount-Doom”*
- *“Q: Where is Bilbo now?”* → *“A: Grey-havens”*
- *“Q: Where is Frodo now?”* → *“A: Shire”*

Why memory? Example!

- *"Bilbo travelled to the cave. Gollum dropped the ring there. Bilbo took the ring. This sentence is random noise for illustration purposes. Bilbo went back to the Shire. Bilbo left the ring there. Frodo got the ring. Frodo journeyed to Mount-Doom. Frodo dropped the ring there. Sauron died. Frodo went back to the Shire. Bilbo travelled to the Grey-havens. The End."*
- *"Q: Where is the ring?"* → *"A: Mount-Doom"*
- *"Q: Where is Bilbo now?"* → *"A: Grey-havens"*
- *"Q: Where is Frodo now?"* → *"A: Shire"*
- Can we design a network that answers such questions?

Memory networks

- Neural network models that
  - have large memory that can store many facts
  - have a learning component for how to read, store, forget and access these facts
- Intuitively, they should work like a “Neural RAM” or a “”Neural Wikipedia”
  - The network processes Wikipedia like information. It needs to store them appropriately for easy read/write/delete/access actions.
  - You make a question
  - The network should recognize the right types of memories
  - The network should reply the question with a meaningful (non trivial) answer.

What is difficult with memory?

- Some sentences are factual
  - “Frodo got the ring”, “Frodo went back to the Shire”
- Some sentences might be random noise
  - *“This sentence is random noice for illustration purposes”*
- To answer a question you might need to combine facts
  - “Where did Frodo get the ring?”
  - *“Bilbo went back to the Shire”* → *“Bilbo left the ring there.”* → *“Frodo got the ring.”*
  - To answer correctly all three sentences need to be carefully analyzed
- TOO MUCH INFORMATION within a single story!!!
  - Can a standard memory unit cope with that?

What is difficult with memory? *(2)*

- Each new story can be completely different
  - Very little data to actually train on
- If we use real data, we don't (usually) have annotations
  - How to analyze mistakes?
- Solution for the last two problems: Make own dataset
  - Start from simple factual sentences and build artificial stories
- Example
  - "John is in the playground.
    Bob is in the office.
    John picked up the football.
    Bob went to the kitchen."
    "Q: Where is the football?" → *"A:playground"*
    "Q: Where was Bob before the kitchen?" "A:office"

Why not simply LSTMs?

- Probably its memory is not large enough
- In latest experiments it seems that LSTMs are not flexible enough for these tasks
  - Although one could maybe create an LSTM-version more specific for the task
- At the end of the day this is still research of the last year
  - *"A research topic that has gained popularity within a small circle of deep learning researchers over the last few months is the combination of a deep neural net and short-term memory. Basically, the neural net acts as a "reasoning" engine that stores and retrieves data to be operated on from a separate memory."*
  - https://www.facebook.com/FBAIResearch/posts/362517620591864 (Nov 3 2014)

Memory Networks

Attributes:
umbrella
beach
sunny
day
people
sand
laying
blue
green
mountain

Internal Textual Representation:
A group of people enjoying a sunny day at the beach with umbrellas in the sand.

External Knowledge:
An umbrella is a canopy designed to protect against rain or sunlight. Larger umbrellas are often used as points of shade on a sunny beach. A beach is a landform along the coast of an ocean. It usually consists of loose particles, such as sand....

Question Answering:
Q: Why do they have umbrellas? **A :** Shade.

Figure 1. A real case of question answering based on an internal textual representation and external knowledge. All of the attributes, textual representation, knowledge and answer are produced by our VQA model. Underlined words indicate the information required to answer the question.

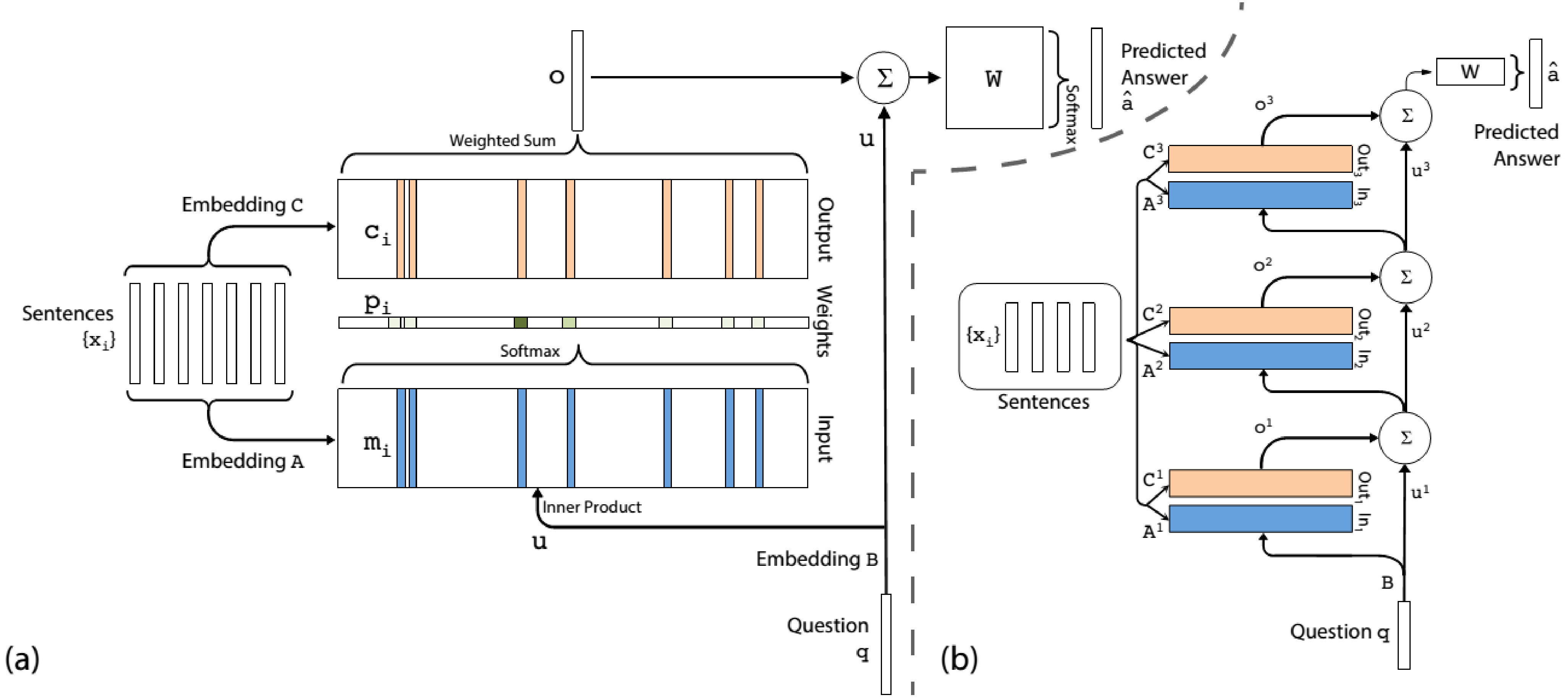
Papers in the literature

- *Neural Turing Machines*, A. Graves, G. Wayne, I. Danihelka, arXiv 2014
  - http://arxiv.org/abs/1410.5401
- *Memory Networks*, J. Weston, S. Chopra, A. Bordes, arXiv 2014
  - http://arxiv.org/abs/1410.3916
- *End-to-end Memory Networks*, S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, arXiv 2015
  - http://arxiv.org/abs/1503.08895
- *Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources,* Q. Wu, P. Wang, C. Shen, A. van den Hengel, A. Dick, arXiv 2015
  - http://arxiv.org/abs/1511.06973

Papers in the literature

- *Neural Turing Machines*, A. Graves, G. Wayne, I. Danihelka, arXiv 2014
  - http://arxiv.org/abs/1410.5401
- *Memory Networks*, J. Weston, S. Chopra, A. Bordes, arXiv 2014
  - http://arxiv.org/abs/1410.3916
- *End-to-end Memory Networks*, S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, arXiv 2015
  - http://arxiv.org/abs/1503.08895
- *Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources,* Q. Wu, P. Wang, C. Shen, A. van den Hengel, A. Dick, arXiv 2015
  - http://arxiv.org/abs/1511.06973

End-to-end Memory Networks



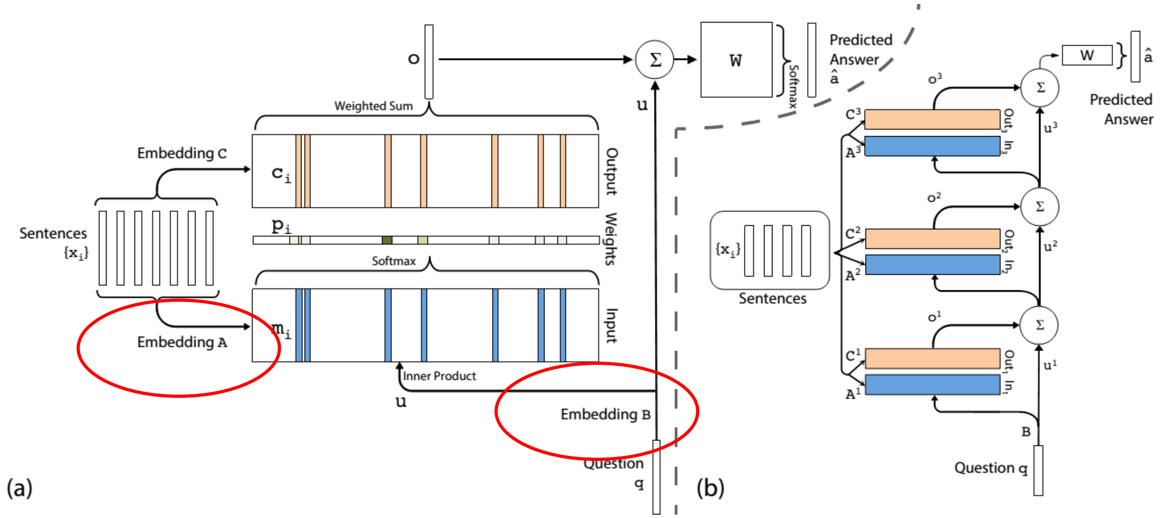
- Torch code available
  - https://github.com/facebook/MemNN/tree/master/MemN2N-lang-model

End-to-end Memory Network unit

- Input memory representation
  - Embeds incoming data to internal representation
- Generalization
  - Given a new input, this unit updates the network memories
- Output
  - Given the memories and given the input, this unit returns a new state variable in the internal representation space of the network
- Response
  - Given the output this unit returns a response recognizable by humans

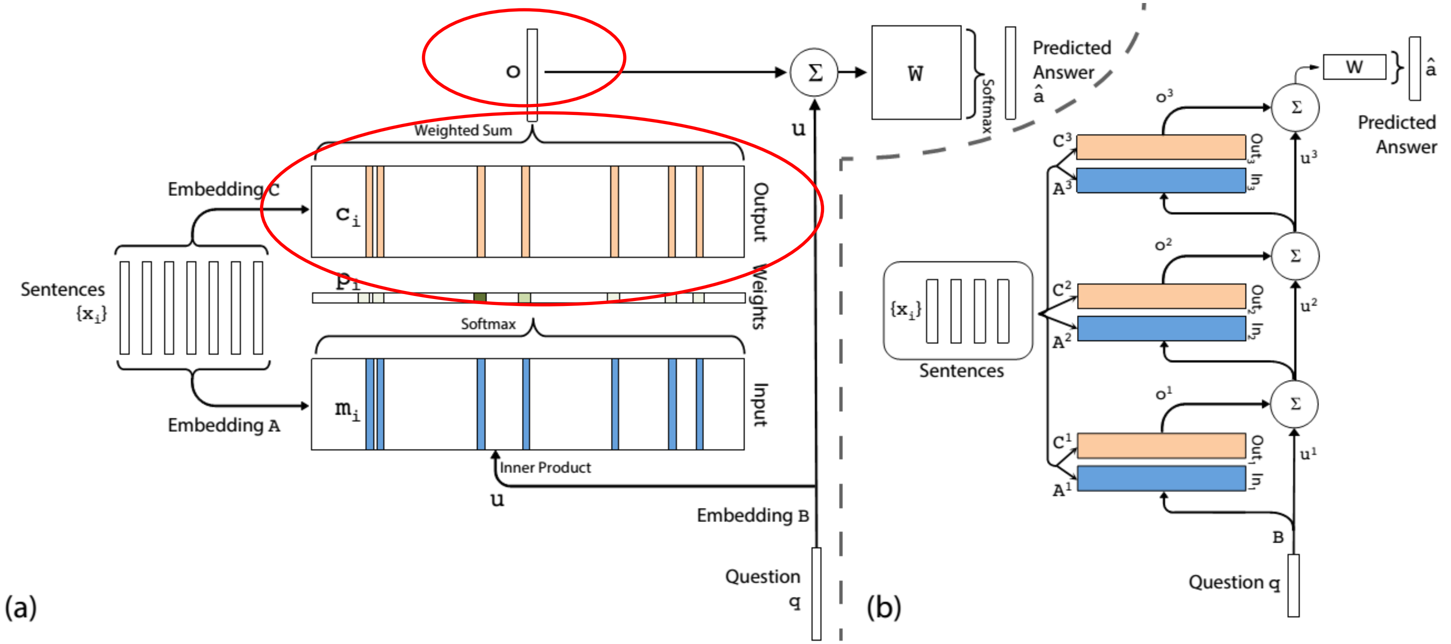
End-to-end Memory Networks: Step (1)

- Input memory representation
- Two embeddings A, B
  - A embeds stories into memory slots on an internal representation space → m_i
  - B embeds the question on the same internal representation space → u
  - To compare memories with questions → $p_i = \text{softmax}(u^T m_i)$



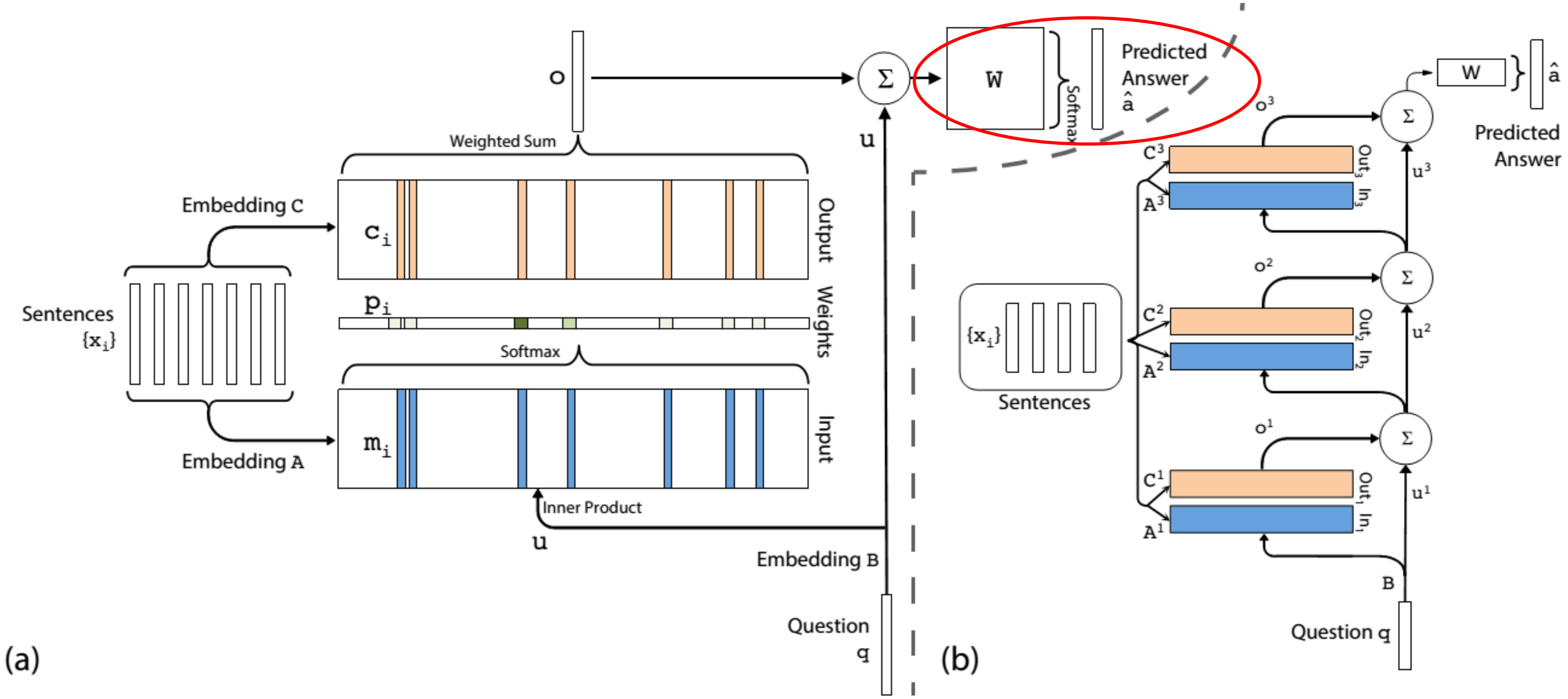
End-to-end Memory Networks: Step (2)

- Output memory representation
  - $o = \sum_i p_i c_i$, where $c_i = C x_i$
- The function that connects the output to the input is smooth
  - Easy to compute gradients for backpropagation



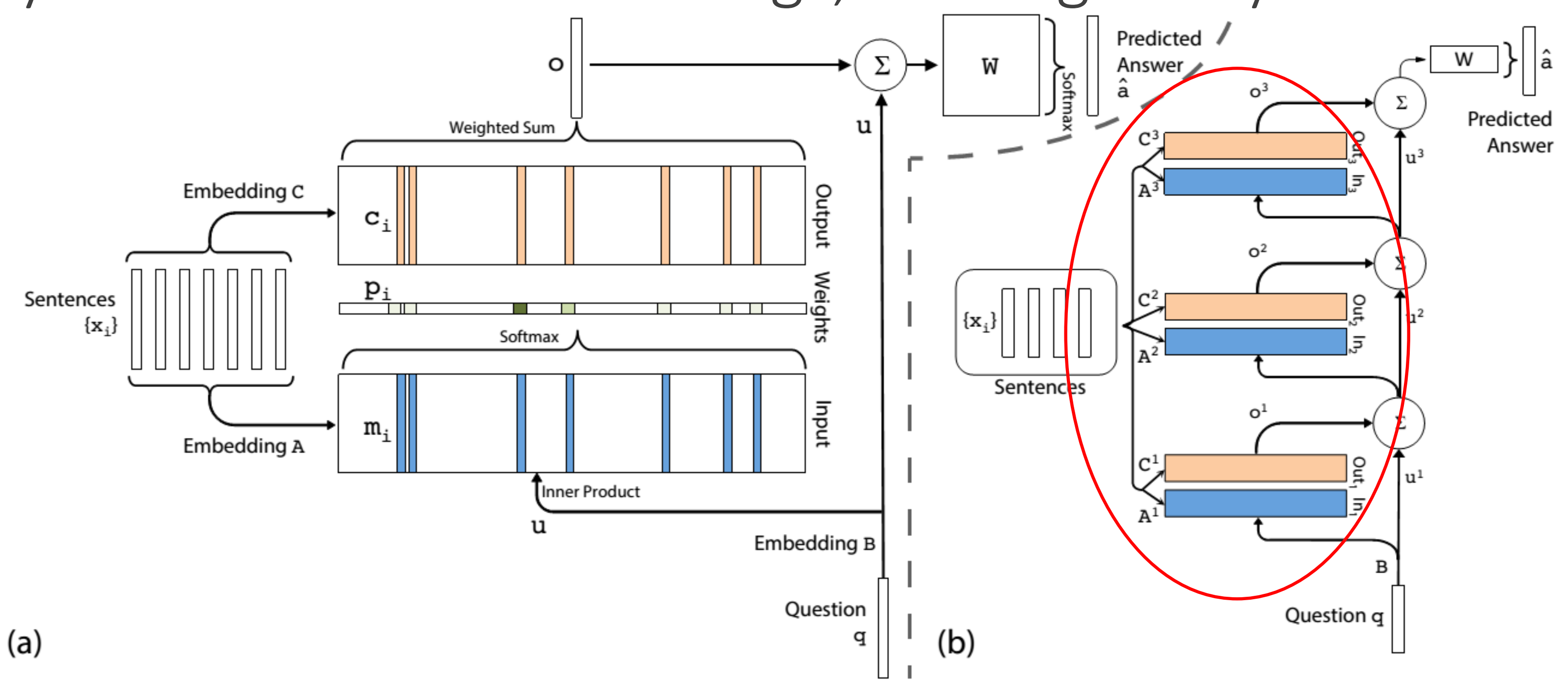
End-to-end Memory Networks: Step (3)

- Final prediction
  - Given the question embedding u and the generated output o
  - $\hat{a} = \mathrm{softmax}(W(o+u))$



End-to-end Memory Networks: Step (4)

- Adding multiple memory layers
  - The input question for a layer is the output plus the question of the previous layer
  - $u^{k+1} = u^k + o^k$
- Each layer has its own embeddings, although they can be tied together



Additional tricks

- Use multiple hops/steps/layers of memories
  - Increases the memory depth of the network
- Use word embeddings and Bag-of-Words representations as inputs
- Use an RNN as response unit
- Add a forget mechanism for when memory is full
- Maybe go to lower level of tokenization
  - Words, letters, chunks
  - Put chunks into memory slots

Successes/failures

Story (15: basic deduction)	Support	Hop 1	Hop 2	Hop 3
Cats are afraid of wolves.	yes	0.00	0.99	0.62
Sheep are afraid of wolves.		0.00	0.00	0.31
Winona is a sheep.		0.00	0.00	0.00
Emily is a sheep.		0.00	0.00	0.00
Gertrude is a cat.	yes	0.99	0.00	0.00
Wolves are afraid of mice.		0.00	0.00	0.00
Mice are afraid of wolves.		0.00	0.00	0.07
Jessica is a mouse.		0.00	0.00	0.00
What is gertrude afraid of? Answer: wolf Prediction: wolf				

Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Lily is a swan.		0.00	0.00	0.00
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is a swan.		0.00	0.00	0.00
Bernhard is yellow.		0.04	0.00	0.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow Prediction: yellow				

Story (17: positional reasoning)	Support	Hop 1	Hop 2	Hop 3
The red square is below the red sphere.	yes	0.37	0.95	0.58
The red sphere is below the triangle.	yes	0.63	0.05	0.43
Is the triangle above the red square? Answer: yes Prediction: no				

Story (18: size reasoning)	Support	Hop 1	Hop 2	Hop 3
The suitcase is bigger than the chest.	yes	0.00	0.88	0.00
The box is bigger than the chocolate.		0.04	0.05	0.10
The chest is bigger than the chocolate.	yes	0.17	0.07	0.90
The chest fits inside the container.		0.00	0.00	0.00
The chest fits inside the box.		0.00	0.00	0.00
Does the suitcase fit in the chocolate? Answer: no Prediction: no				

What can’t be done, what comes next?

- Current networks answer rather simple questions. Make questions harder
  - “Q: Who is teaching the Deep Learning Course?” → “A. Efstratios Gavves **and** Patrick Putzky”
- Use multiple supporting memories
- More extensive knowledge databases
- More realistic questions and answers
- Perhaps perform actions instead of answers

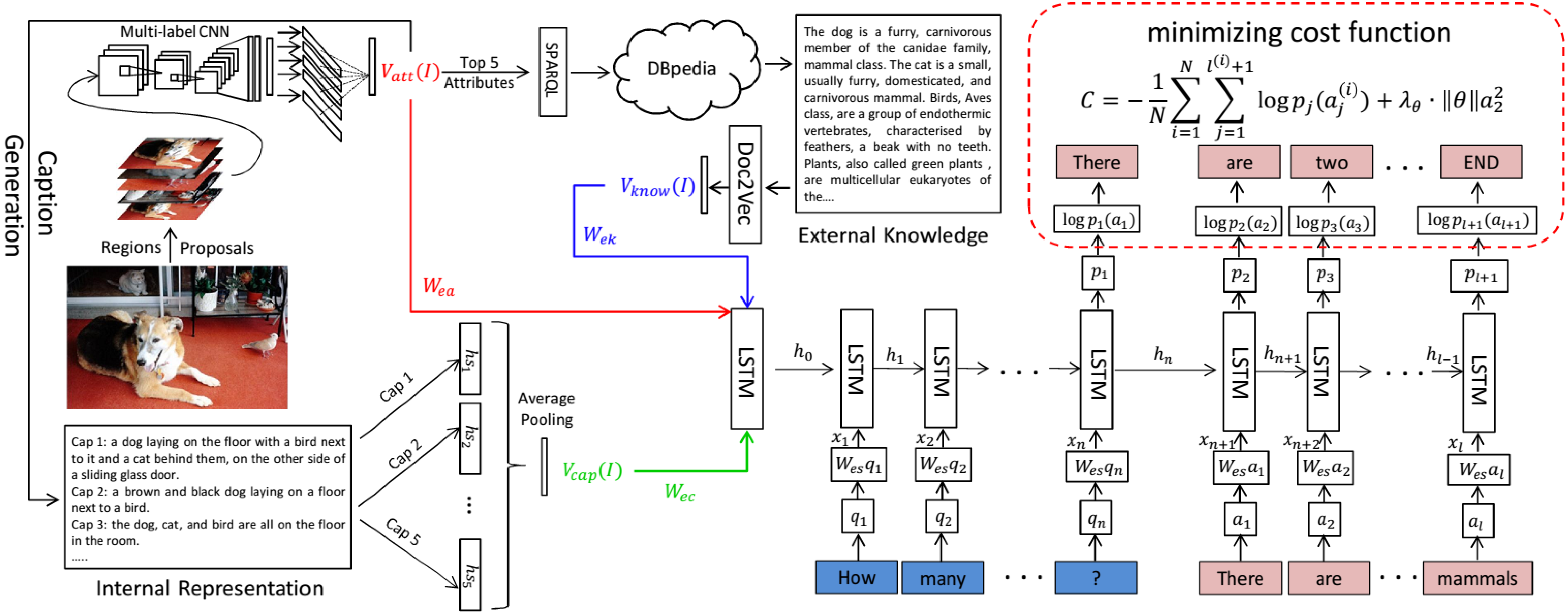
Papers in the literature

- *Neural Turing Machines*, A. Graves, G. Wayne, I. Danihelka, arXiv 2014
  - http://arxiv.org/abs/1410.5401
- *Memory Networks*, J. Weston, S. Chopra, A. Bordes, arXiv 2014
  - http://arxiv.org/abs/1410.3916
- *End-to-end Memory Networks*, S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, arXiv 2015
  - http://arxiv.org/abs/1503.08895
- *Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources*, Q. Wu, P. Wang, C. Shen, A. van den Hengel, A. Dick, arXiv 2015
  - http://arxiv.org/abs/1511.06973

Papers in the literature

- *Neural Turing Machines*, A. Graves, G. Wayne, I. Danihelka, arXiv 2014
  - http://arxiv.org/abs/1410.5401
- *Memory Networks*, J. Weston, S. Chopra, A. Bordes, arXiv 2014
  - http://arxiv.org/abs/1410.3916
- *End-to-end Memory Networks*, S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, arXiv 2015
  - http://arxiv.org/abs/1503.08895
- *Ask Me Anything: Free-form Visual Question Answering Based on Knowledge from External Sources*, Q. Wu, P. Wang, C. Shen, A. van den Hengel, A. Dick, arXiv 2015
  - http://arxiv.org/abs/1511.06973

External databases for visual questions



Method

- Classifying objects, attributes inside an image
  - Use a very deep, VGG-16 network fine-tuned on the MSCOCO image attributes
- Caption-based image representations
  - Use an LSTM
  - Start with an image
  - Generate a caption
  - Use the hidden state h_T of the final step as a representation
- Relate to external database
  - DBpedia
  - SQL-like queries using SPARQL
  - Represent returned text with Doc2Vec
- Combine everything and end-to-end learning
  - $x = [W_{ea} x_{att}(I),\ W_{ec} x_{cap}(I), W_{ek} x_{know}(I),]$

Results




	Why is she wearing a crown?	Why is he smiling?	Why is the zebra on the ground?	Why do they have umbrellas?
Ours:	birthday	happy	resting	shade
Vgg+LSTM:	to eat	unknown	eat	raining
Ground Truth:	birthday	happy	resting	shade




	Why is a man sitting under an umbrella?	Why are there animals pinned to the wall?	Why do they have umbrellas?	Why is he swinging backhand?
Ours:	shade	decoration	raining	to hit ball
Vgg+LSTM:	safety	teddy	yes	tennis ball
Ground Truth:	shade	decoration	raining	to hit ball

More results




	Why do these sheep have paint on them?	Why is his arm outflung?	Why are the animals laying here?	Why are all the giraffes gathered together?
Ours:	identification	balance	resting	eating
Vgg+LSTM:	to eat	to play	no	to play
Ground Truth:	identification	balance	resting	eating




	Why are they wearing such bright colors?	Why are the men wearing orange?	Why is the man jumping?	Why is this room warm?
Ours:	safety	team	skateboarding	fireplace
Vgg+LSTM:	yes	to	unknown	to sleep
Ground Truth:	safety	team	skateboarding	fireplace

Summary

- Memory networks
- Difficulties with modelling memory
- Memory networks for image-language reasoning

Next lecture

- Student presentations of Deep Learning papers

The Recursive Neural Network

Deep Learning Lecture

Sara Veldhoen MSc

March 3, 2016

Outline

The Model

- Variations
- Unsupervised

My Research

- Logic
- Arithmetic
- Analysis

- Goal: model semantics of linguistic utterances
- Lexical distributional semantics: successful
- What about composition?
- How to deal with variable (sentence) length?
- Sequence Models: recurrent connections as memory. All the work is done by a single cell.

- Observation: language is structured
- Compositionality: meaning of a complex expression is a function of its parts and the way they are (syntactically) combined
- Symbolic implementation: Montague Grammar
- Distributional implementation: Recursive Neural Network (RNN) Socher *et al.* (2012)

Outline

The Model

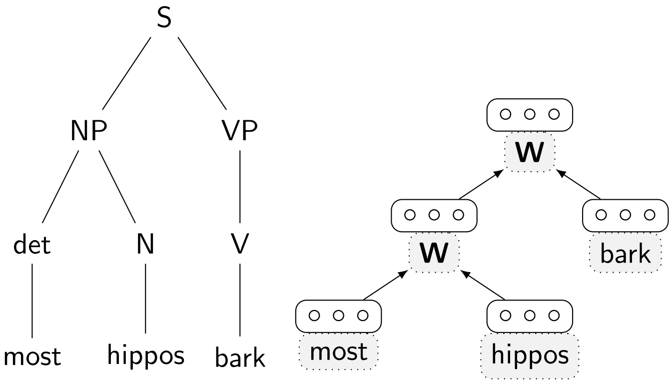
- Variations
- Unsupervised

My Research

- Logic
- Arithmetic
- Analysis

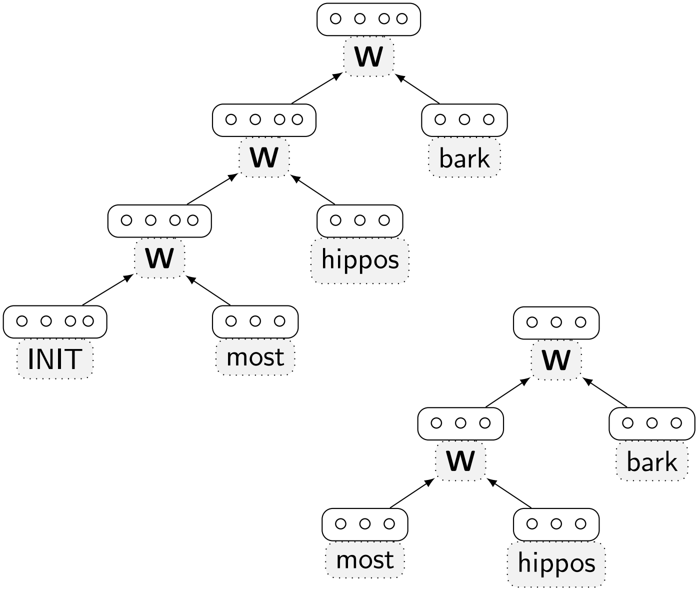
The Model

Recursive Neural Network:



$$\mathbf{p} = f(\mathbf{W} \cdot [\mathbf{c}_0; \mathbf{c}_1] + \mathbf{b})$$

The Model



The Model - Variations

Matrix-Vector (Socher *et al.* , 2012)

$$\mathbf{p}_{MV} = f(\mathbf{W}_v[\mathbf{C}_1\mathbf{c}_2; \mathbf{C}_2\mathbf{c}_1]) \tag{1}$$

$$\mathbf{P}_{MV} = \mathbf{W}_M[\mathbf{C}_1; \mathbf{C_2}] \tag{2}$$

The Model - Variations

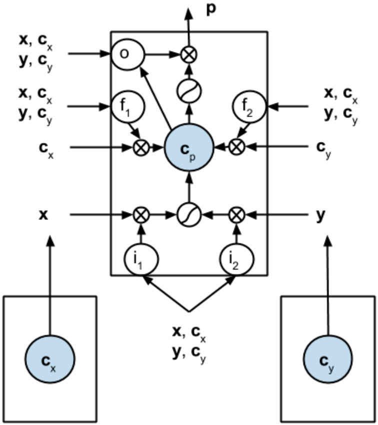
Tensor (Bowman & Potts, 2015)

$$\mathbf{p} = f(\mathbf{W} \cdot [\mathbf{c}_0; \mathbf{c}_1] + \mathbf{b}) \tag{3}$$

$$\mathbf{p}_{MV} = \mathbf{p} + f(\mathbf{c}_1^T \mathbf{T}^{[1 \ldots n]} \mathbf{c}_2) \tag{4}$$

The Model - Variations

Tree-LSTM (Le & Zuidema, 2015)

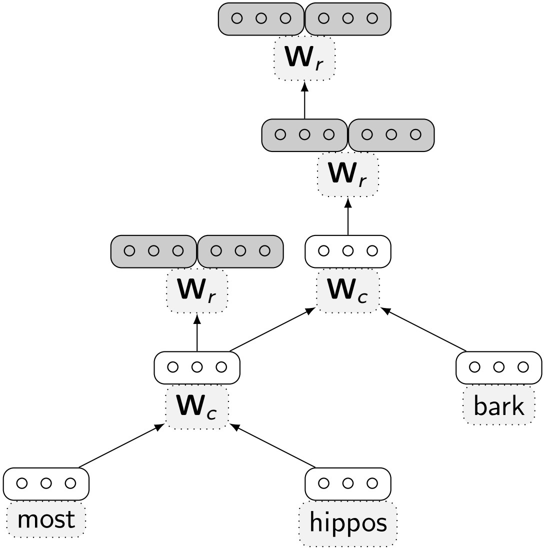


The Model - Variations

Deep composition (Socher *et al.* , 2010)
Add more layers between children and parent representation

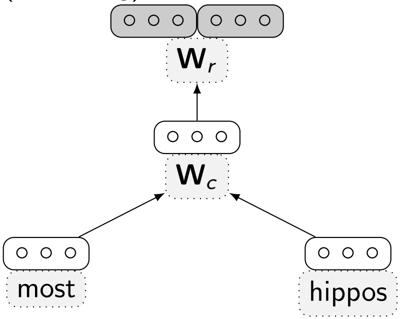
The Model - Unsupervised

(Unfolding) Recursive Auto-Encoder (Socher *et al.* , 2011)



The Model - Unsupervised

(Unfolding) Recursive Auto-Encoder (Socher *et al.* , 2011)

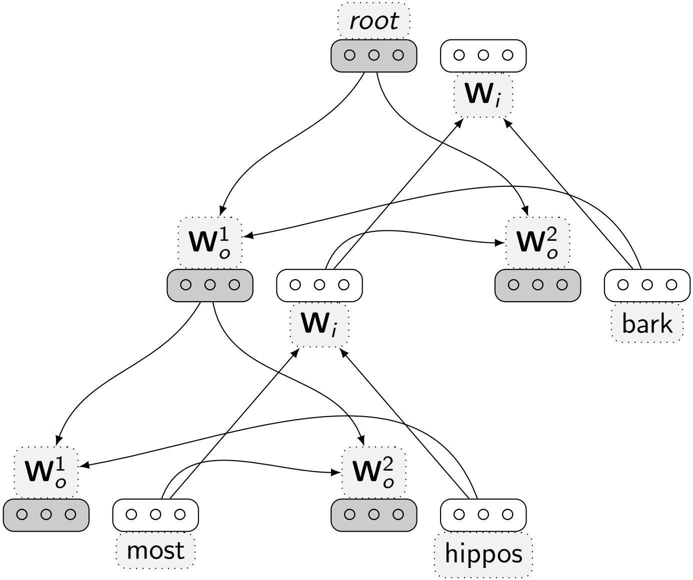


$$\mathbf{p} = f(\mathbf{W_c} \cdot [\mathbf{c}_0; \mathbf{c}_1] + \mathbf{b_c}) \tag{5}$$

$$[\mathbf{c}_0^r; \mathbf{c}_1^r] = f(\mathbf{W_r} \cdot \mathbf{p} + \mathbf{b_r}) \tag{6}$$

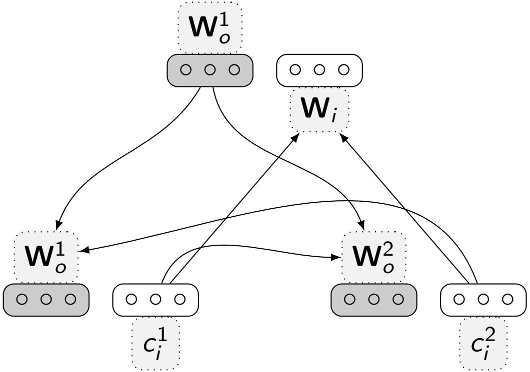
The Model - Unsupervised

Inside-Outside RNN (Le & Zuidema, 2014)



The Model - Unsupervised

Inside-Outside RNN (Le & Zuidema, 2014)



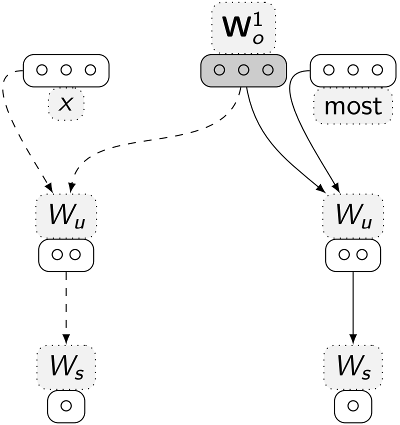
$$\mathbf{p_i} = f(\mathbf{W_i} \cdot [\mathbf{c}_i^1; \mathbf{c}_i^2] + \mathbf{b_i}) \tag{7}$$

$$\mathbf{c_o^1} = f(\mathbf{W_o^1} \cdot [\mathbf{p}_o; \mathbf{c}_i^2] + \mathbf{b_o^1}) \tag{8}$$

$$\mathbf{c_o^2} = f(\mathbf{W_o^1} \cdot [\mathbf{p}_o; \mathbf{c}_i^1] + \mathbf{b_o^2}) \tag{9}$$

The Model - Unsupervised

Inside-Outside RNN (Le & Zuidema, 2014) training:



$$c(w, x) = \max\{0, 1 - s(w, \mathbf{o_w}) + s(x, \mathbf{o_w})\} \qquad (10)$$

$$s(x, \mathbf{o_w}) = \mathbf{W_s} f(\mathbf{W_u}[\mathbf{o_w}; \mathbf{i_x}] + \mathbf{b_u}) \qquad (11)$$

Outline

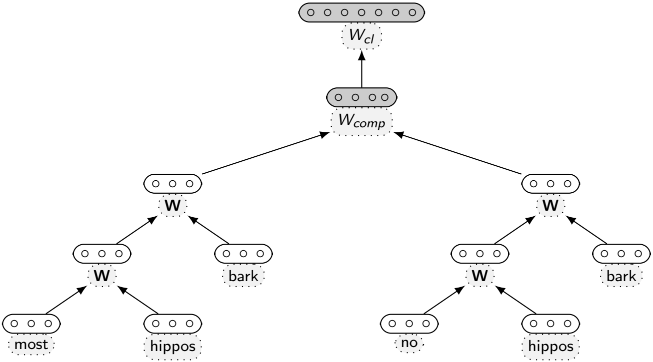
My Research - Logic

Bowman & Potts (2015) train RNN directly for NLI

- Train RNN through classifier
- Data: pairs of sentences with inference relation

(most hippo) bark	$\mid$	(no hippo) bark
(most hippo) bark	$\sqsupset$	(no hippo) (not bark)
(two hippo) bark	$\#$	(some (not hippo)) bark
(three hippo) Parisian	$\sqsubset$	(three hippo) French
(all hippo) Parisian	$\sqsubset$	(all hippo) French

My Research - Logic



My Research - Logic

Replication of Bowman & Potts (2015) and extension

	2014 Data Composition		2015 Data Composition	
	Fixed	Trained	Fixed	Trained
Fixed Embs	45.9	99.7	29.7	97.3
Trained Embs	84.8	99.6	58.2	92.9

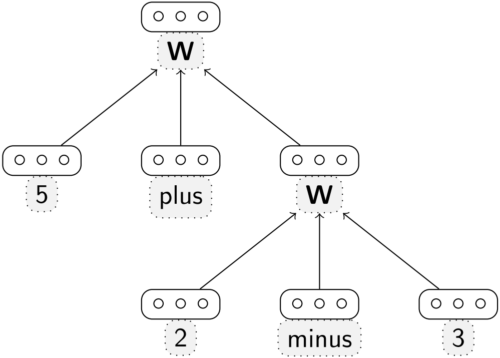
Table: The effect of fixing the word embeddings or the composition function. Accuracy (%) on the test data.

Does the model actually capture logical semantics?

My Research - Arithmetic

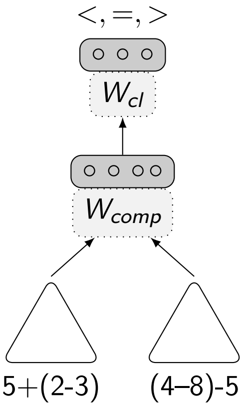
Simple task: arithmetic expression trees

Principled solution captures sense of numbers: *numerosity*



My Research - Arithmetic

Set-up: comparison layer and soft-max classifier on top of two trees



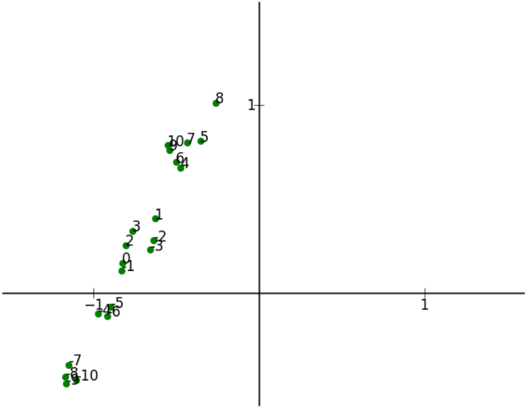
My Research - Arithmetic

Setting		Dimensionality 1	2	10
trained embs	trained comp	49.8	88.2*	97.0
	fixed comp	50.3	51.6	63.2
fixed embs	trained comp	49.0	50.7	68.5
	fixed comp	47.2	49.6	50.5

Table: Accuracy (%) on held-out data. *: the variance over different runs was less than one percentage point in all cases but one: the 2 dimensional setting had one run performing considerably worse than the others; 63.1% accuracy vs. an average of 96.5% for the rest.

My Research - Arithmetic

- The RNN can learn to do addition and subtraction, and decide which of two expressions is greater.
- What has the model learned? Is it a principled solution?
- 2-D case: can be plotted.



My Research - Analysis

Step-by-step analysis: Project-Sum-Squash

- Break up the composition function:

$$\mathbf{p} = f(\mathbf{W} \cdot [\mathbf{c}_0; \mathbf{c}_1; \mathbf{c}_2] + \mathbf{b})$$

$$\mathbf{p} = f(\mathbf{W}_0\mathbf{c}_0 + \mathbf{W}_1\mathbf{c}_1 + \mathbf{W}_2\mathbf{c}_2 + \mathbf{b})$$

- The intermediate results can be plotted

My Research - Analysis

Breaking up the composition function

$$\begin{pmatrix} a & b & c & d & e & f \\ g & h & i & j & k & l \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

$$= \begin{pmatrix} a1 + b2 + c3 + d4 + e5 + f6 \\ g1 + h2 + i3 + j4 + k5 + l6 \end{pmatrix}$$

$$= \begin{pmatrix} a1 + b2 \\ g1 + h2 \end{pmatrix} + \begin{pmatrix} c3 + d4 \\ i3 + j4 \end{pmatrix} + \begin{pmatrix} e3 + f4 \\ k3 + l4 \end{pmatrix}$$

$$= \begin{pmatrix} a & b \\ g & h \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \end{pmatrix} + \begin{pmatrix} c & d \\ i & j \end{pmatrix} \times \begin{pmatrix} 3 \\ 4 \end{pmatrix} + \begin{pmatrix} e & f \\ k & l \end{pmatrix} \times \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

My Research - Analysis

Breaking up the composition function

$$\begin{pmatrix} a & b & c & d & e & f \\ g & h & i & j & k & l \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

$$= \begin{pmatrix} a1 + b2 + c3 + d4 + e5 + f6 \\ g1 + h2 + i3 + j4 + k5 + l6 \end{pmatrix}$$

$$= \begin{pmatrix} a1 + b2 \\ g1 + h2 \end{pmatrix} + \begin{pmatrix} c3 + d4 \\ i3 + j4 \end{pmatrix} + \begin{pmatrix} e3 + f4 \\ k3 + l4 \end{pmatrix}$$

$$= \begin{pmatrix} a & b \\ g & h \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \end{pmatrix} + \begin{pmatrix} c & d \\ i & j \end{pmatrix} \times \begin{pmatrix} 3 \\ 4 \end{pmatrix} + \begin{pmatrix} e & f \\ k & l \end{pmatrix} \times \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

My Research - Analysis

Breaking up the composition function

$$\begin{pmatrix} a & b & c & d & e & f \\ g & h & i & j & k & l \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

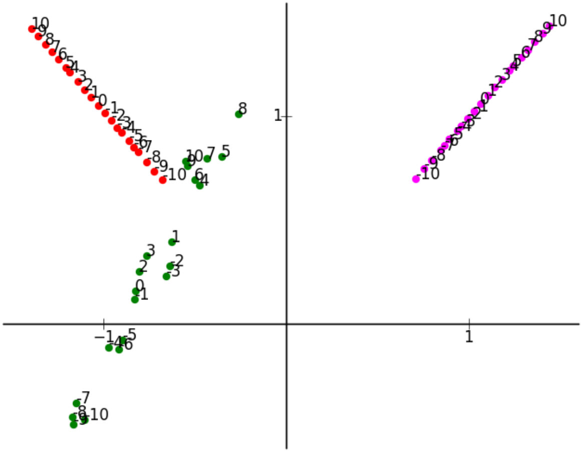
$$= \begin{pmatrix} a1 + b2 + c3 + d4 + e5 + f6 \\ g1 + h2 + i3 + j4 + k5 + l6 \end{pmatrix}$$

$$= \begin{pmatrix} a1 + b2 \\ g1 + h2 \end{pmatrix} + \begin{pmatrix} c3 + d4 \\ i3 + j4 \end{pmatrix} + \begin{pmatrix} e3 + f4 \\ k3 + l4 \end{pmatrix}$$

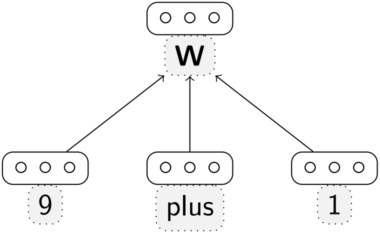
$$= \begin{pmatrix} a & b \\ g & h \end{pmatrix} \times \begin{pmatrix} 1 \\ 2 \end{pmatrix} + \begin{pmatrix} c & d \\ i & j \end{pmatrix} \times \begin{pmatrix} 3 \\ 4 \end{pmatrix} + \begin{pmatrix} e & f \\ k & l \end{pmatrix} \times \begin{pmatrix} 5 \\ 6 \end{pmatrix}$$

My Research - Analysis

Lexical embeddings and projections

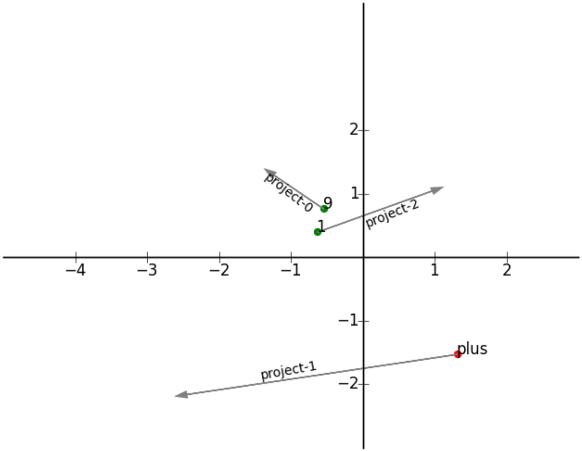


My Research - Example: 9+1



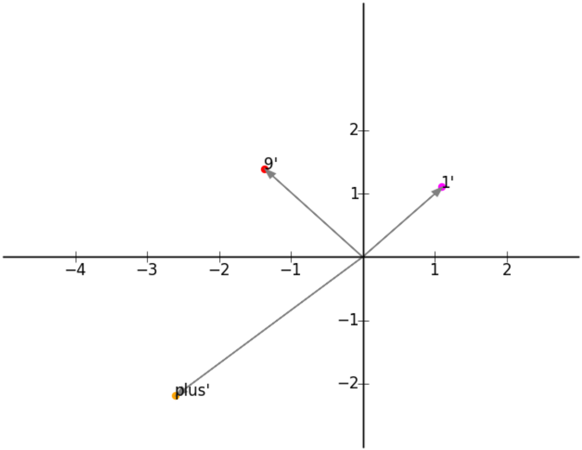
My Research - Example: 9+1

Project



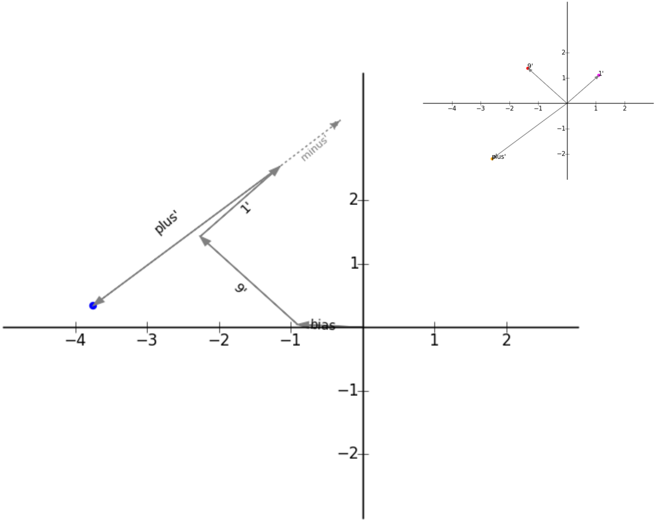
My Research - Example: 9+1

Project



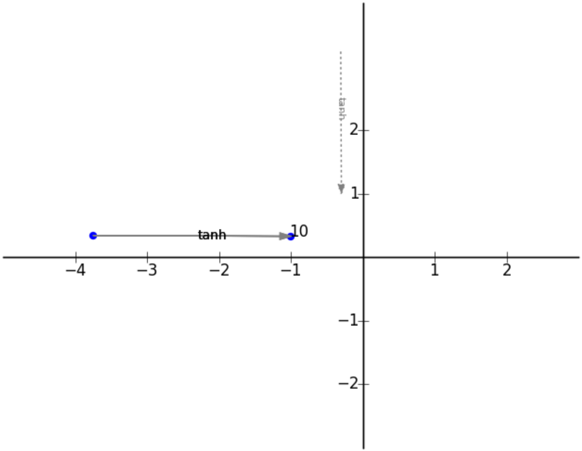
My Research - Example: 9+1

Sum



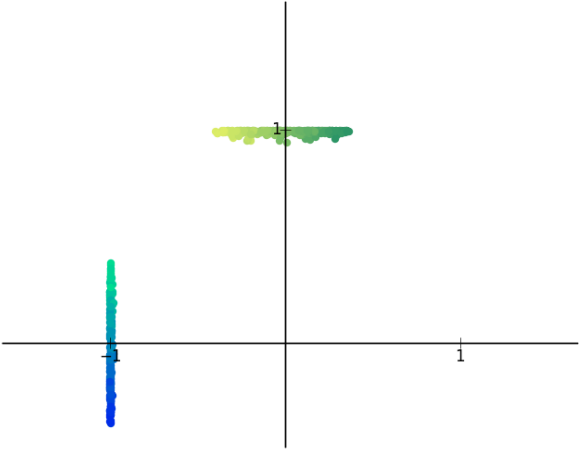
My Research - Example: 9+1

Squash



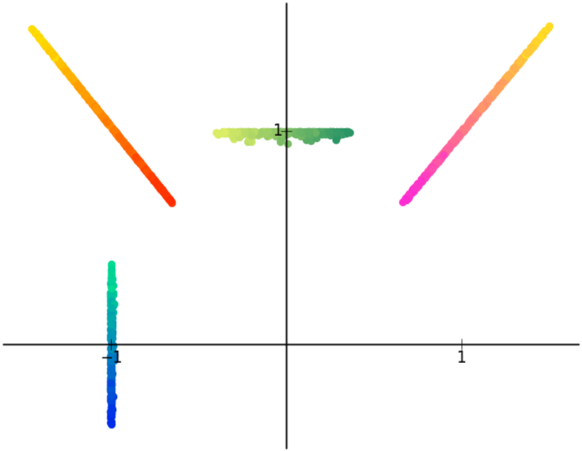
My Research -

Parent representations



My Research -

Parent representations and projections



My Research -

- Project-Sum-Squash provides information on how the model fulfills a task
  Interpretation: is it a principled solution?
- The same technique can be applied to the higher dimensional case
  One needs dimensionality reduction, e.g. PCA

My Research - Future Work

- Can the RNN really learn logical reasoning?
- Compare different composition functions
- Unsupervised training
- Language generation from sentence representations
- Reduce reliance on syntactic parse

References

Bowman, Samuel R, & Potts, Christopher. 2015. Recursive neural networks can learn logical semantics. *ACL-IJCNLP 2015*, 12.

Le, Phong, & Zuidema, Willem. 2014. Inside-Outside Semantics: A Framework for Neural Models of Semantic Composition.

Le, Phong, & Zuidema, Willem. 2015. Compositional distributional semantics with long short term memory. *arXiv preprint arXiv:1503.02510*.

Socher, Richard, Manning, Christopher D, & Ng, Andrew Y. 2010. Learning continuous phrase representations and syntactic parsing with recursive neural networks. *Pages 1–9 of: Proceedings of the NIPS-2010 Deep Learning and Unsupervised Feature Learning Workshop*.

Socher, Richard, Huang, Eric H, Pennin, Jeffrey, Manning, Christopher D, & Ng, Andrew Y. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. *Pages 801–809 of: Advances in Neural Information Processing Systems*.

Socher, Richard, Huval, Brody, Manning, Christopher D, & Ng, Andrew Y. 2012. Semantic compositionality through recursive matrix-vector spaces. *Pages 1201–1211 of: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics.