

# Neural Network Tools and Principles, part 3

Sequential modeling, recurrent neural networks

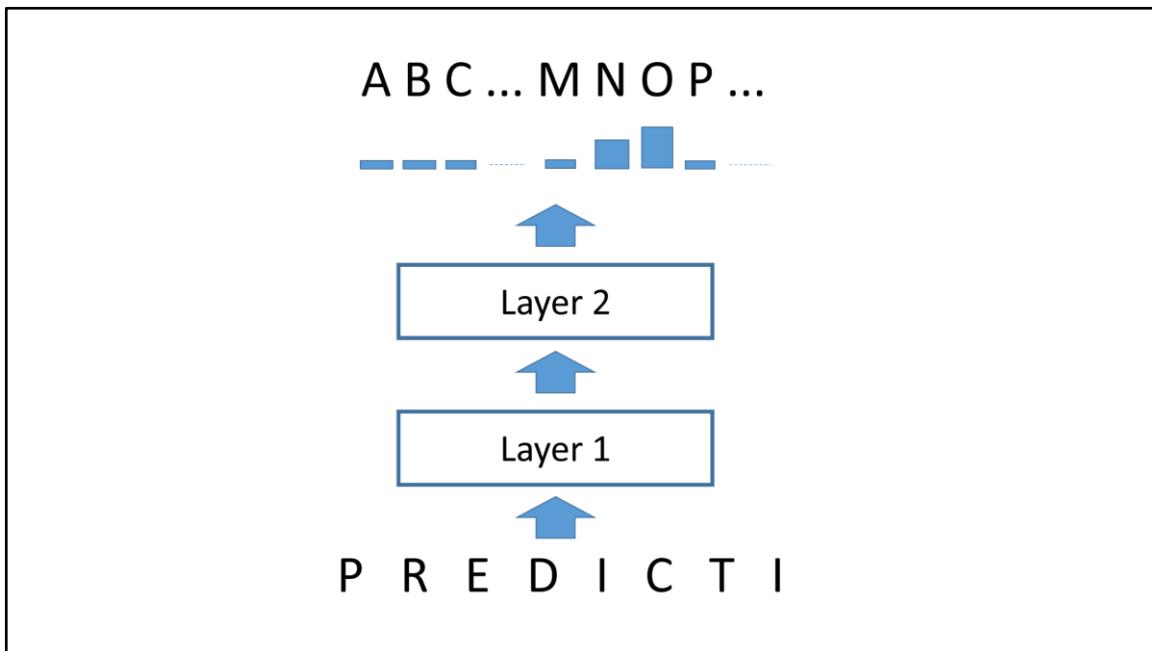
Intelligent Computational Media, Spring 2020

Prof. Perttu Hämäläinen

Aalto University

## Sequence modeling

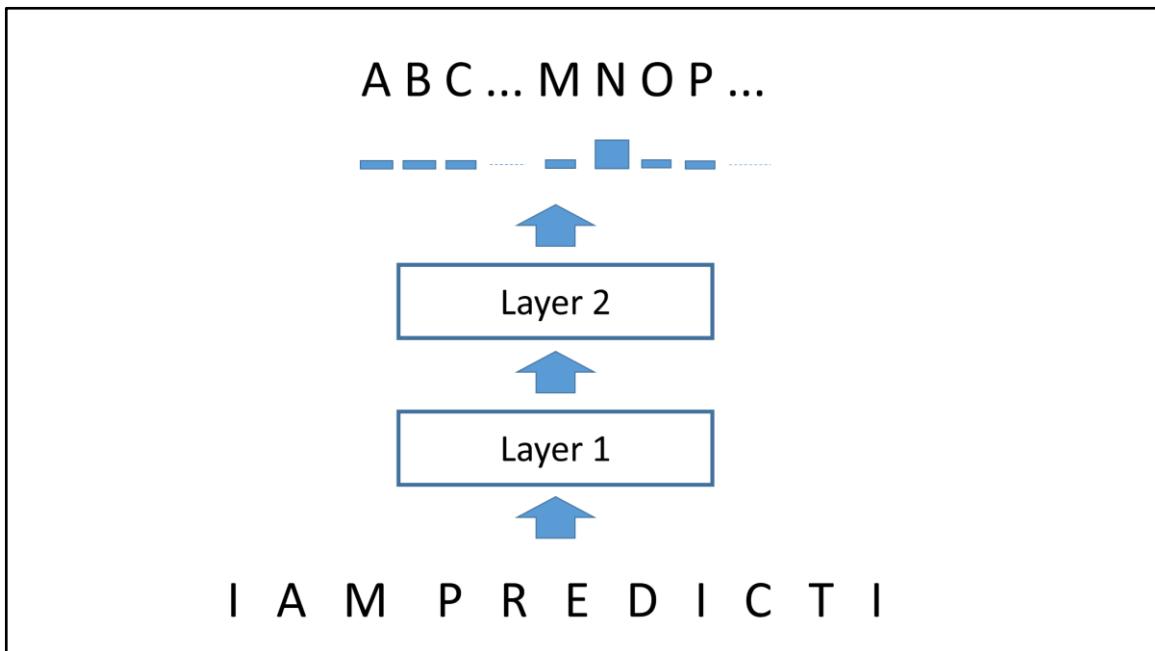
- Predictive text input
- Generating text
- Generating audio
- Language translation (sequence-to-sequence mapping)



Simple example: predict the next character, or the probabilities of possible next characters.

This could be trained with the basic tools we've covered so far: just have a standard classification network with an output neuron for each possible character, and train with softmax cross-entropy loss, using random 8-character input snippets from some text data.

This can then be used in two ways: pick the most probable character, or randomly sample a character based on the probabilities. Now, if one shifts the input sequence to the left and adds the sampled character to the right, one can use this kind of a network for text generation.



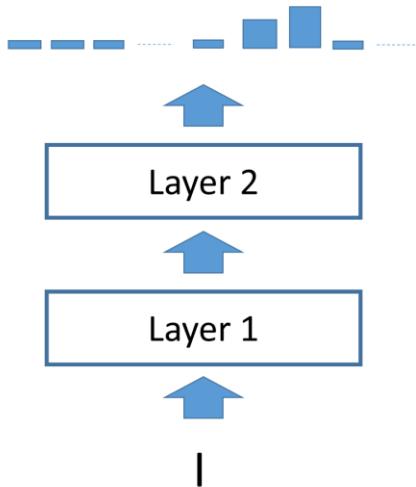
The problem is, however, to select the correct input sequence length.

In principle, one wants infinite input sequences, as this gives the network more contextual understanding.

For example, if one wants to generate a novel, one must understand context not just within a sentence, but within paragraphs, pages and chapters.

Infinite input sequence length = infinite computing cost

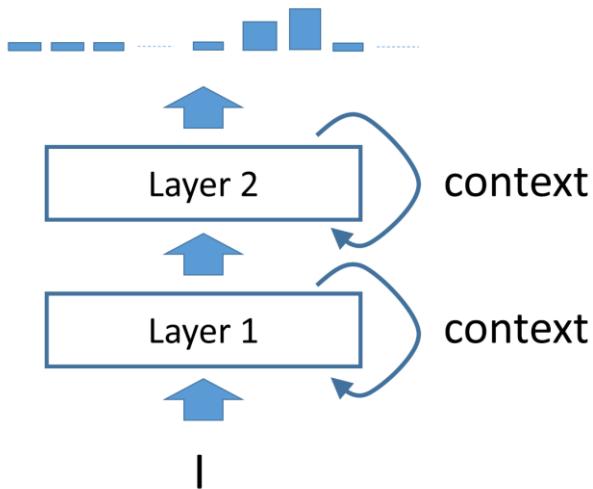
A B C ... M N O P ...



How to make the network build and maintain context internally, when trained one symbol at a time?

It would thus be desirable

A B C ... M N O P ...



Recurrent networks to the rescue: add feedback connections (to be used as additional inputs for the next processed symbol)

## Unrolling the compute graph

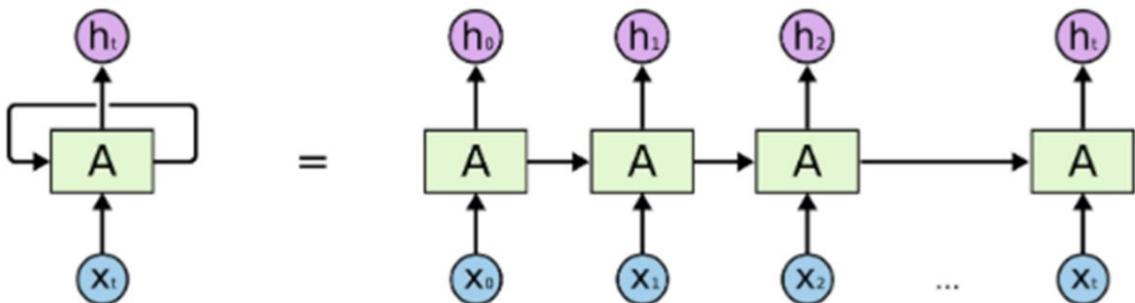


Image source: <https://medium.com/@camrongodbout/recurrent-neural-networks-for-beginners-7aca4e933b82>

It should be noted that the recurrent compute graph on the left is equivalent to the "unrolled" compute graph on the right.

This is key to training recurrent networks: during training, the compute graph is unrolled, which allows backpropagating gradients. The unrolled graph simply has many copies of the same network, each with shared parameters so that the optimization does not try to change them to different values.

## What can it do?

- <http://karpathy.github.io/2015/05/21/rnn-effectiveness/>
- Tensorflow version: <https://github.com/sherjilozair/char-rnn-tensorflow/blob/master/model.py>
- The same as a Colab notebook:  
<https://colab.research.google.com/drive/13Vr3PrDg7cc4OZ3W2-grLSVSf0RJYWzb>

Naturalism and decision for the majority of Arab countries' capitalide was grounded by the Irish language by [[John Clair]], [[An Imperial Japanese Revolt]], associated with Guangzham's sovereignty. His generals were the powerful ruler of the Portugal in the [[Protestant Immineners]], which could be said to be directly in Cantonese Communication, which followed a ceremony and set inspired prison, training. The emperor travelled back to [[Antioch, Perth, October 25|21]] to note, the Kingdom of Costa Rica, unsuccessful fashioned the [[Thrales]], [[Cynth's Dajoard]], known in western [[Scotland]], near Italy to the conquest of India with the conflict. Copyright was the succession of independence in the slop of Syrian influence that was a famous German movement based on a more popular servitious, non-doctrinal and sexual power post. Many governments recognize the military housing of the [[Civil Liberalization and Infantry Resolution 265 National Party in Hungary]], that is sympathetic to be to the [[Punjab Resolution]] (PJS) [<http://www.humah.yahoo.com/guardian>.

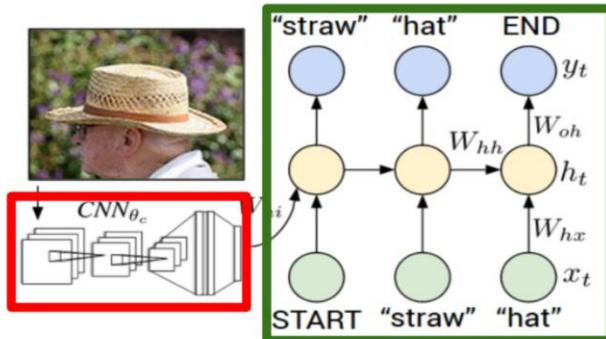
cfm/7754800786d17551963s89.htm Official economics Adjoint for the Nazism, Montgomery was swear to advance to the resources for those Socialism's rule, was starting to signing a major tripad of aid exile.]

## Character-level RNN trained on Wikipedia

## What can it do?

- <https://cs.stanford.edu/people/karpathy/sfmltalk.pdf>

### Recurrent Neural Network



### Convolutional Neural Network

Combination of convolutional neural networks and recurrent neural networks can be trained to describe images in natural language.

The key here is that both network types can be added to the same compute graph to allow end-to-end training with backpropagation. The convolutional network first builds an abstract high-level representation of the image input, and this is used to initialize the hidden state of the text-generating RNN.

a man standing in front of a store with a woman in the background



Here's an example of running such network real-time on a laptop's webcam while walking around the city.

[Submitted on 16 Nov 2018]

## Generating Albums with SampleRNN to Imitate Metal, Rock, and Punk Bands

CJ Carr, Zack Zukowski

This early example of neural synthesis is a proof-of-concept for how machine learning can drive new types of music software. Creating music can be as simple as specifying a set of music influences on which a model trains. We demonstrate a method for generating albums that imitate bands in experimental music genres previously unrealized by traditional synthesis techniques (e.g. additive, subtractive, FM, granular, concatenative). Raw audio is generated autoregressively in the time-domain using an unconditional SampleRNN. We create six albums this way. Artwork and song titles are also generated using materials from the original artists' back catalog as training data. We try a fully-automated method and a human-curated method. We discuss its potential for machine-assisted production.

Comments: 3 pages

Subjects: **Sound (cs.SD)**; Audio and Speech Processing (eess.AS)

Journal reference: Proceedings of the 6th International Workshop on Musical Metacreation (MUME 2018)

Cite as: arXiv:1811.06633 [cs.SD]

(or arXiv:1811.06633v1 [cs.SD] for this version)

<http://dadabots.com>, also see their YouTube channel, e.g., <https://youtu.be/EoqBhkB5oI0?t=102>

Audio can also be generated similar to text, using a recurrent model that generates new outputs based on its previous outputs.

## RNN benefits

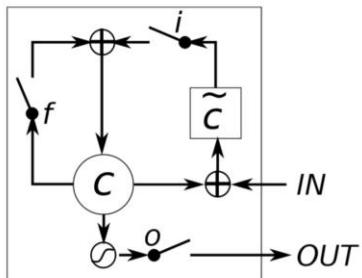
- Input sequence length not fixed
- More computationally efficient for long sequences in runtime

## RNN Problems

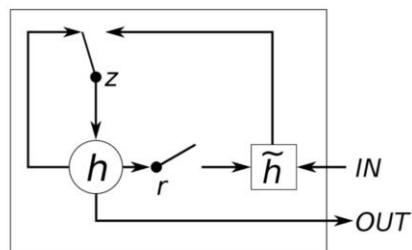
- Training may be unstable. How to init internal state for each unrolled sequence?
- Longer unrolling improves results but uses more memory
- Very long memory does not emerge in practice

## LSTM, GRU

- Solve the long-term memory problem using *gates*



(a) Long Short-Term Memory



(b) Gated Recurrent Unit

Figure 1: Illustration of (a) LSTM and (b) gated recurrent units. (a)  $i$ ,  $f$  and  $o$  are the input, forget and output gates, respectively.  $c$  and  $\tilde{c}$  denote the memory cell and the new memory cell content. (b)  $r$  and  $z$  are the reset and update gates, and  $h$  and  $\tilde{h}$  are the activation and the candidate activation.

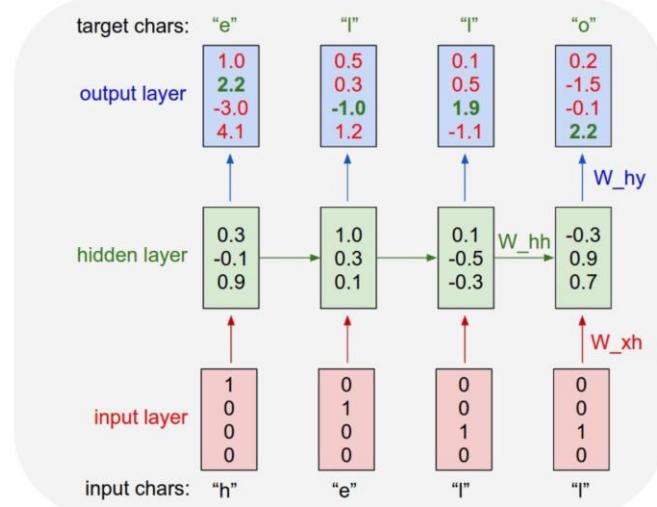
Image source: <https://arxiv.org/pdf/1412.3555.pdf>

Basic principle: just replace each recurrent neuron with this kind of more complex LSTM or GRU unit. Readymade implementations exist in Tensorflow etc.

The gates allow the network to learn when to update the contextual information and when to forget.

GRU is more recent and usually provides similar performance with a lower computing cost due to each GRU only requiring two gates.

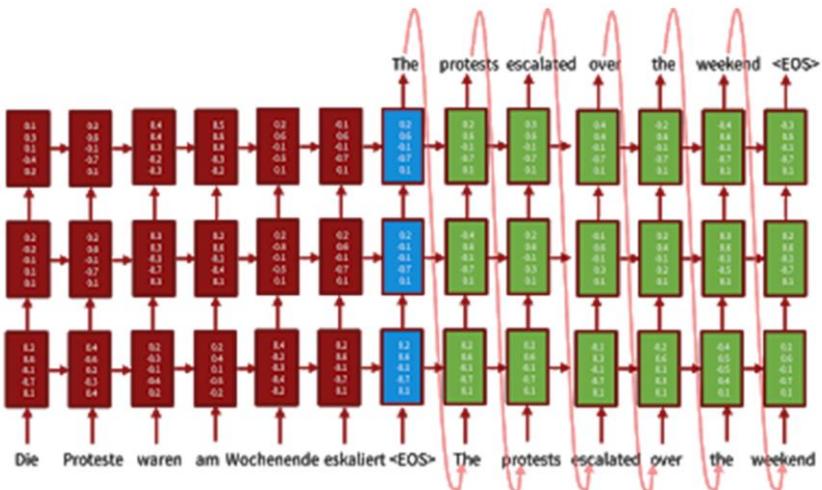
## More details: one-hot encoding



Note that input symbols are typically encoded with one-hot representation, i.e., vectors of length N for an N character alphabet.

An alternative is to use word embeddings like the ones discussed in the previous talk.

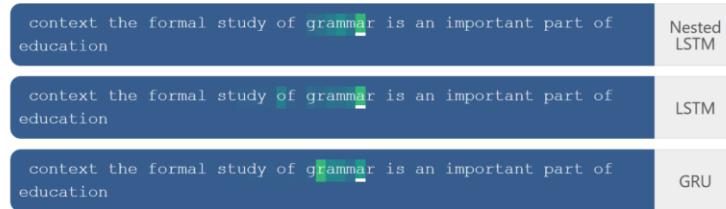
## Neural machine translation



One can also have each word as it's own symbol, like in this

# Visualizing memorization in RNNs

Inspecting gradient magnitudes in context can be a powerful tool to see when recurrent units use short-term or long-term contextual understanding.



This **connectivity visualization** shows how strongly previous input characters influence the current target character in an [autocomplete problem](#). For example, in the prediction of "grammar" the GRU RNN [initially](#) uses long-term memorization but as [more characters become available](#) the RNN switches to short-term memorization. ([reset](#))

---

AUTHORS

Andreas Madsen

AFFILIATIONS

NearForm Ltd

PUBLISHED

March 25, 2019

DOI

10.23915/distill.00016

---

<https://distill.pub/2019/memorization-in-rnns/>

A nice article that shows how and when RNN:s use past inputs

## Advanced

- <https://distill.pub/2016/augmented-rnns/> (Neural Turing machines etc.)

## Case study: Relation networks

- Combining convolutional image processing and LSTM or GRU to allow answering natural language questions about images

# A simple neural network module for relational reasoning

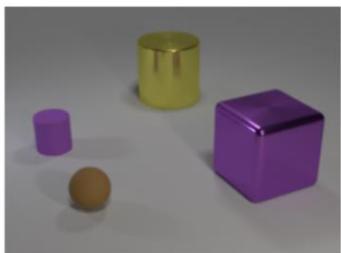
Santoro, Raposo, Barrett, Malinowski, Pascanu, Battaglia, Lillicrap

Submitted to arxiv on 5 Jun 2017

This summary by Perttu Hämäläinen

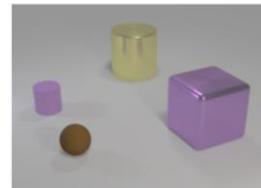
I'm skipping this on the lecture to save time, and it's too involved if you're just interested in using deep learning instead of developing deep learning systems. However, it's a fascinating system and I'm keeping this material in the lecture slides for those interested.

**Original Image:**



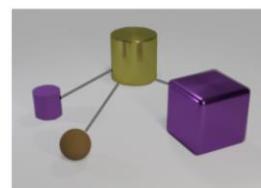
**Non-relational question:**

What is the size of  
the brown sphere?



**Relational question:**

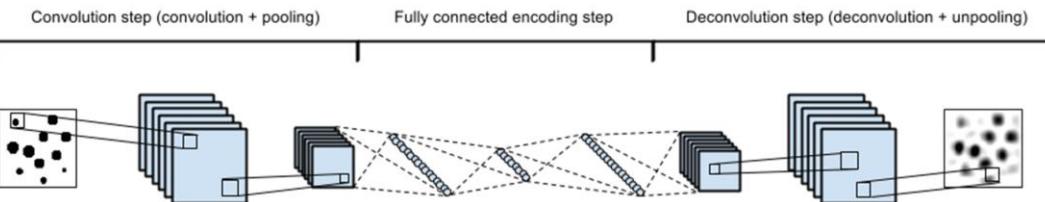
Are there any rubber  
things that have the  
same size as the yellow  
metallic cylinder?



This image from the paper gives an example of what type of problems we are discussing today. This is obviously related to human-computer interaction, e.g., general question answering, virtual assistants and recommender systems. This kind of relational questions have remained an open problem, but this paper appears to solve it for now, achieving superhuman performance on most existing benchmarks.

## How to represent relations?

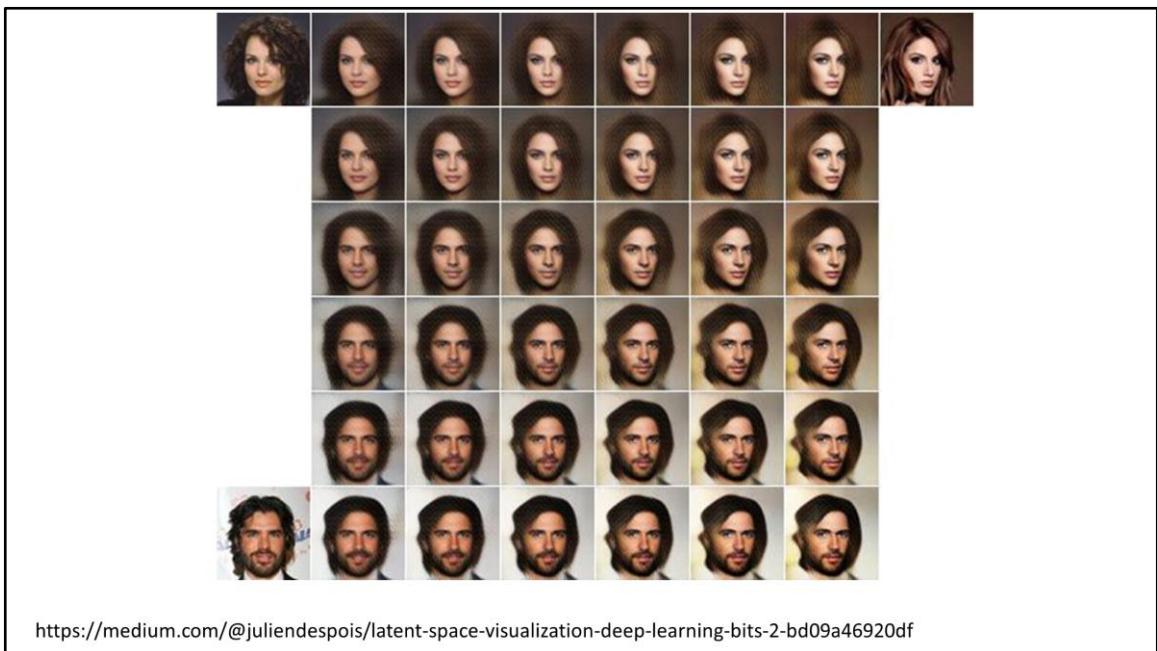
- In standard network architectures, unit output represents presence/absence/strength of a feature (line, corner, eye, face, femininity/masculinity, bald vs hair)
- Outputs of all units of a layer represent the input in some feature space (latent space)
- But how to go from features to relations, especially in complex inputs such as images?



<https://swarbrickjones.wordpress.com/2015/04/29/convolutional-autoencoders-in-python-theano-lasagne/>

In convolutional autoencoders, for example, further layers lose spatial resolution in exchange for more abstract & compact features.

“The latent feature space in the middle contains a **compressed** representation of the image, which is **the only information** the decoder is allowed to use to try to reconstruct the input **as faithfully as possible**. To perform well, the network has to learn to extract the **most relevant** features in the bottleneck.”



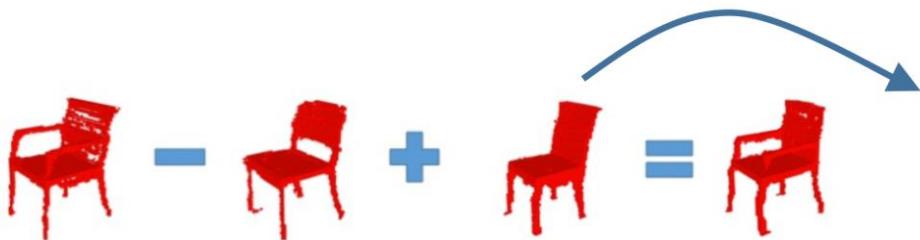
<https://medium.com/@juliendespois/latent-space-visualization-deep-learning-bits-2-bd09a46920df>

This allows for interesting tricks such as interpolation in the feature space (here, a 2D latent space of faces constructed by a convolutional autoencoder)



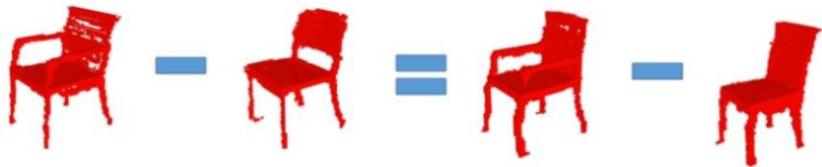
Wu et al., Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling, NIPS 2016

...and even simple arithmetic. Here, adding and subtracting feature vectors has surprisingly interpretable results.



Wu et al., Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling, NIPS 2016

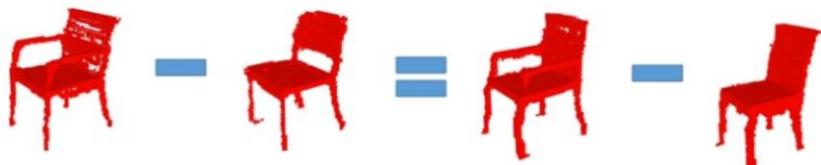
Now, note that if the normal algebraic rules hold in this pictorial equation, we should be able to move this chair to the other side by changing its sign.



Wu et al., Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling, NIPS 2016

Now, as subtracting two vectors results in a vector defining the relation of the two vectors in space, we can read out the equation in a relational form

This is to that like this is to that



Wu et al., Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling, NIPS 2016

## How to answer questions about relations?

We've so far seen that standard architectures learn something about relations, or at least relations can be probably deduced from the feature representations. However, a CNN alone is not good in answering relational questions. It appears we need something more.

## How to answer questions about relations?

- In many problems, selecting the correct network architecture is key – the architecture imposes a strong prior on the learning
- Convnets: same features/structures can occur anywhere spatially, spatial connectivity is sparse
- This paper: what kind of architecture allows the units to represent relations efficiently?
- Answer: Relational Network (RN)

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$

This is the formula for the output of a relational network or RN

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$

Objects



Here, the  $o$  denote objects, i.e., some input features that are assumed to be interpretable as objects.

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$

MLPs      Objects

The  $f$  and  $g$  are both MLP networks

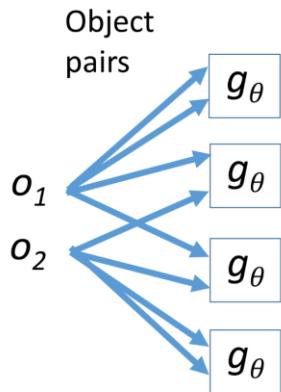
$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$

$o_1$

$o_2$

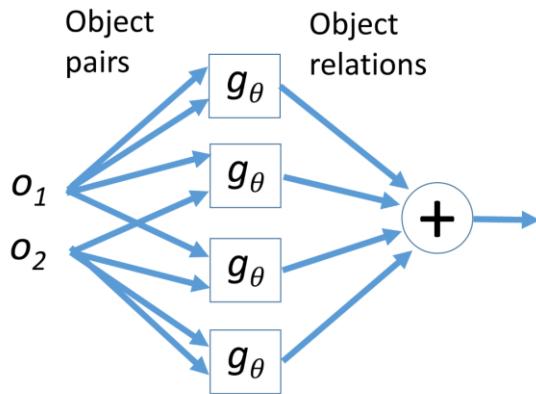
Let's unpack this by visualizing the full compute graph in the simple case when there are only two "objects"

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$



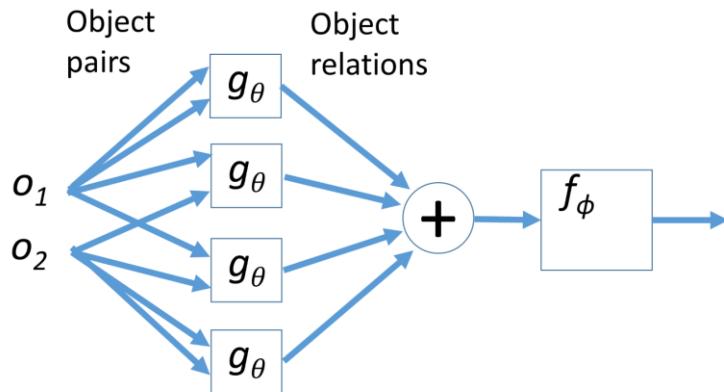
The summation over pairs of  $i, j$  means that there are several copies (square of the number of inputs) of the "g" network, which all share the same parameters.

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$



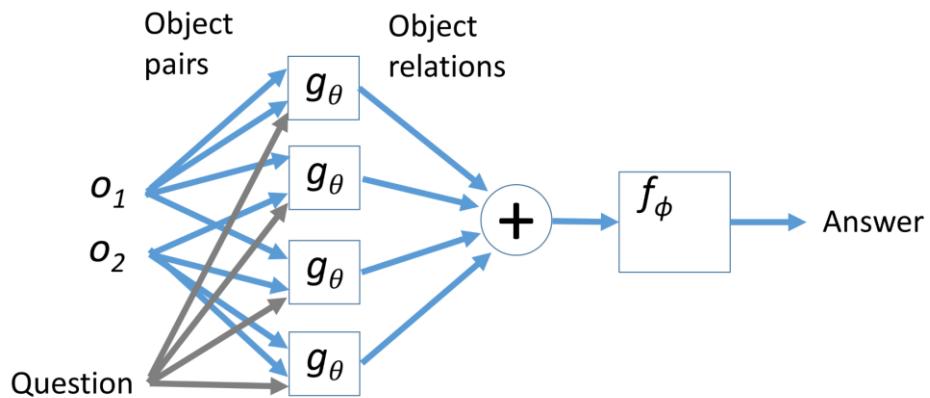
The outputs of all copies are then added together, which forms a kind of summary of object relations.

$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$

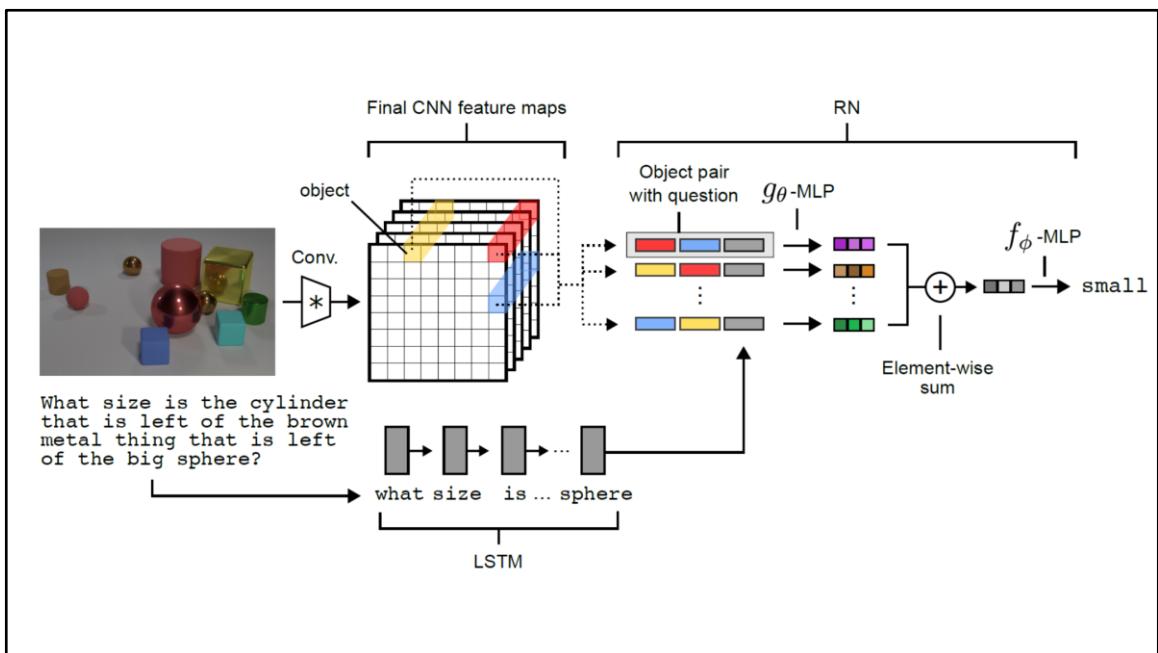


The summary is then fed to the "f" network which transforms it to something. However, this something is not yet the answer to any question.

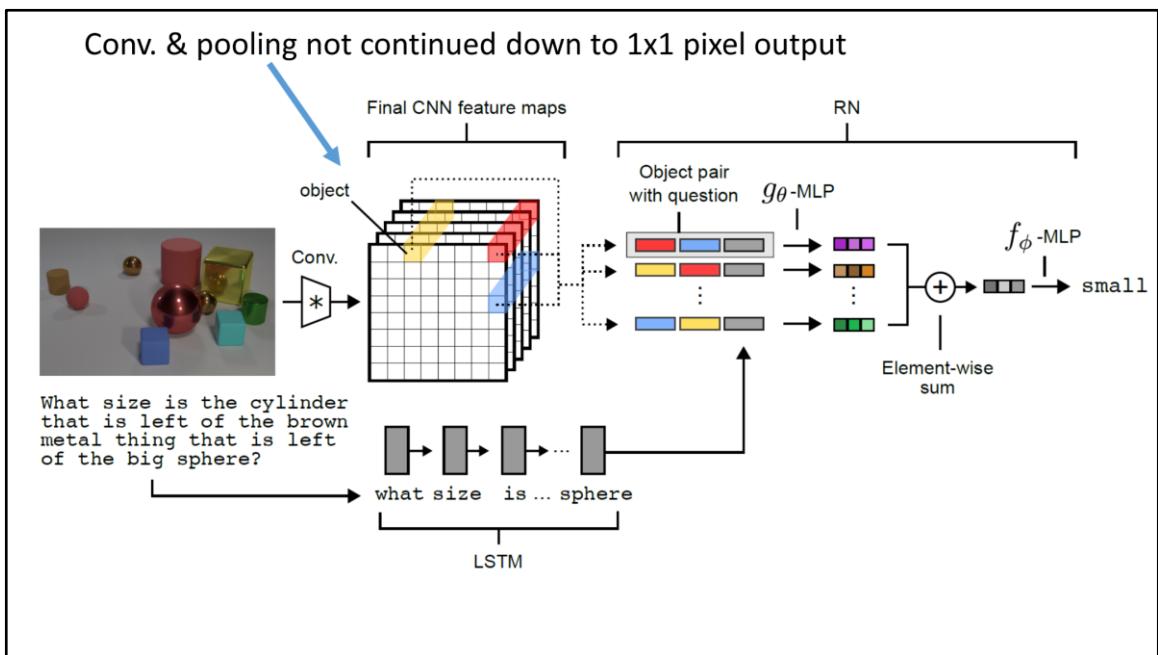
$$\text{RN}(O) = f_\phi \left( \sum_{i,j} g_\theta(o_i, o_j) \right)$$



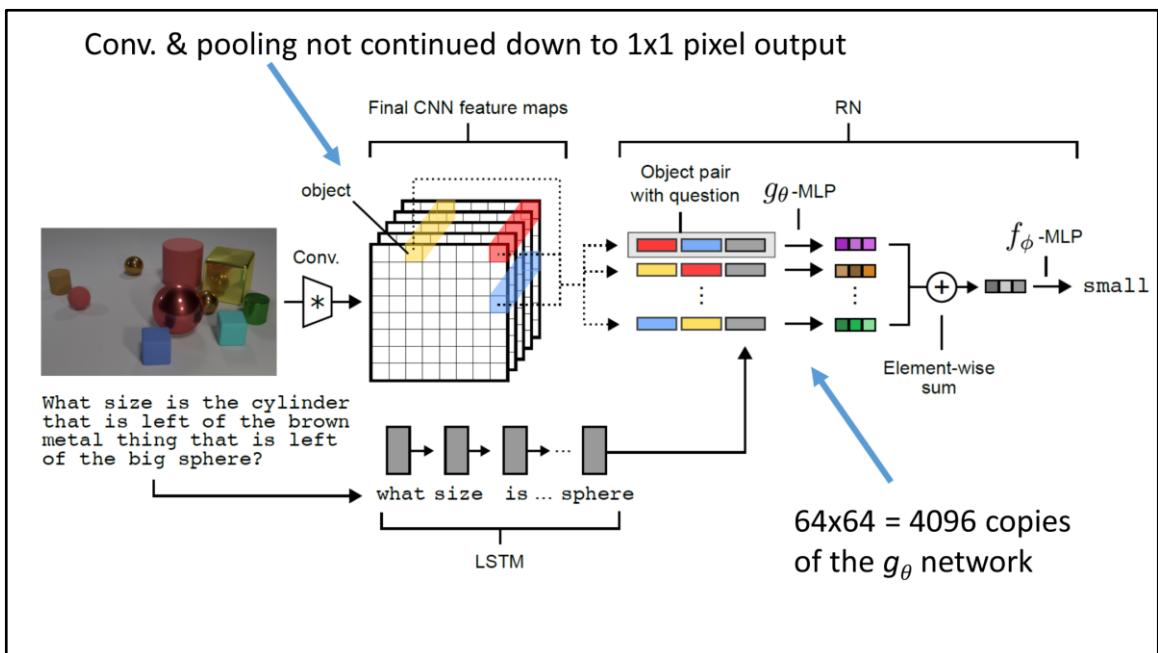
In their experiments, the authors encode the questions into vectors and feed the question vector to the "g" modules with the object pairs. When trained with a question answering dataset, the output then becomes the answer.



Here's a more complete architecture picture from the paper. It's remarkable that the RN module seems to be quite flexible in how it can be connected with any networks that output features and question encodings. In this picture, visual question answering uses a standard convnet to encode image objects and an LSTM to encode the questions.



Note that to preserve spatial information, the convolution hierarchy ends before the output size is reduced to a single pixel. Here, 8x8 pixel outputs are used, meaning that the network can basically distinguish between at maximum 64 objects that are visible at the same time



Note that handling all object pairs means that there is a quite large amount of copies of the "g" network in the full compute graph. Of course, they share parameters, but objects pairs and gradients still need to be fed through them, i.e., it's not the computationally most lightweight architecture.

## Benchmarks

- CLEVR dataset with 14 tasks (image-based question answering)
- Sort-of-CLEVR (separated relational and non-relational questions)
- bAbI (facebook research text understanding and reasoning)
- Dynamic physical systems (infer invisible springs between some moving objects)
- Superhuman performance in over 90% of the benchmarks

bAbI example: Mary moved to the bathroom. John went to the hallway. Where is Mary?

bAbI also contains many more types of benchmarks, e.g., dialogue between a bot and a customer.

Model	Overall	Count	Exist	Compare Numbers	Query Attribute	Compare Attribute
Human	92.6	86.7	96.6	86.5	95.0	96.0
Q-type baseline	41.8	34.6	50.2	51.0	36.0	51.3
LSTM	46.8	41.7	61.1	69.8	36.8	51.8
CNN+LSTM	52.3	43.7	65.2	67.1	49.3	53.0
CNN+LSTM+SA	68.5	52.2	71.1	73.5	85.3	52.3
CNN+LSTM+SA*	76.6	64.4	82.7	77.4	82.6	75.4
<b>CNN+LSTM+RN</b>	<b>95.5</b>	<b>90.1</b>	<b>97.8</b>	<b>93.6</b>	<b>97.9</b>	<b>97.1</b>

\* Our implementation, with optimized hyperparameters and trained fully end-to-end.

Table 1: **Results on CLEVR from pixels.**

This table shows how RN achieves a step change over previous systems.

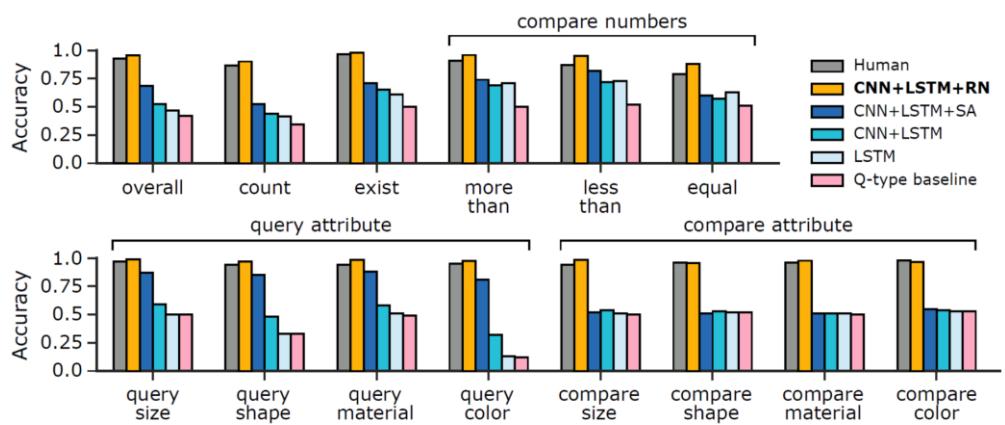
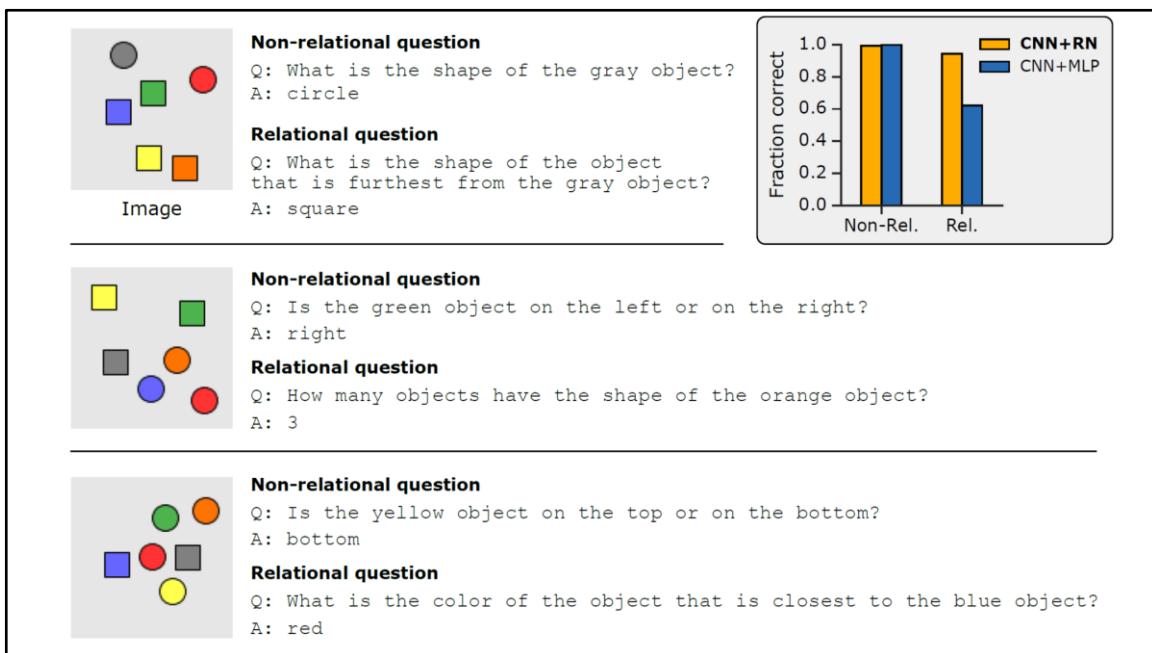
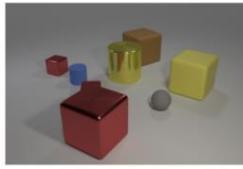


Figure 3: Results on CLEVR from pixels.



Some more examples



What shape is the small object that is in front of the yellow matte thing and behind the gray sphere?

RN: cylinder

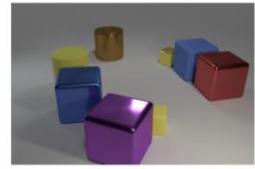
1

2

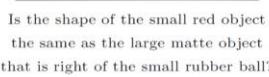
GT: cube

2

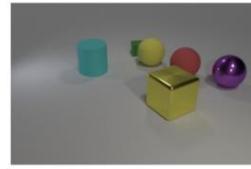
3



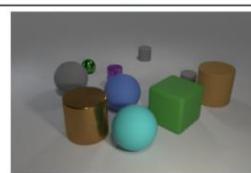
What number of objects are blocks that are in front of the large red cube or green balls?



Is the shape of the small red object the same as the large matte object that is right of the small rubber ball?



What number of things are either tiny green rubber objects or shiny things that are behind the big metal block?



How many gray objects are in front of the tiny green shiny ball and right of the big blue matte thing?

0

5

RN: no

GT: yes

1

6



What number of objects are big red matte cubes or things on the right side of the large red matte block?

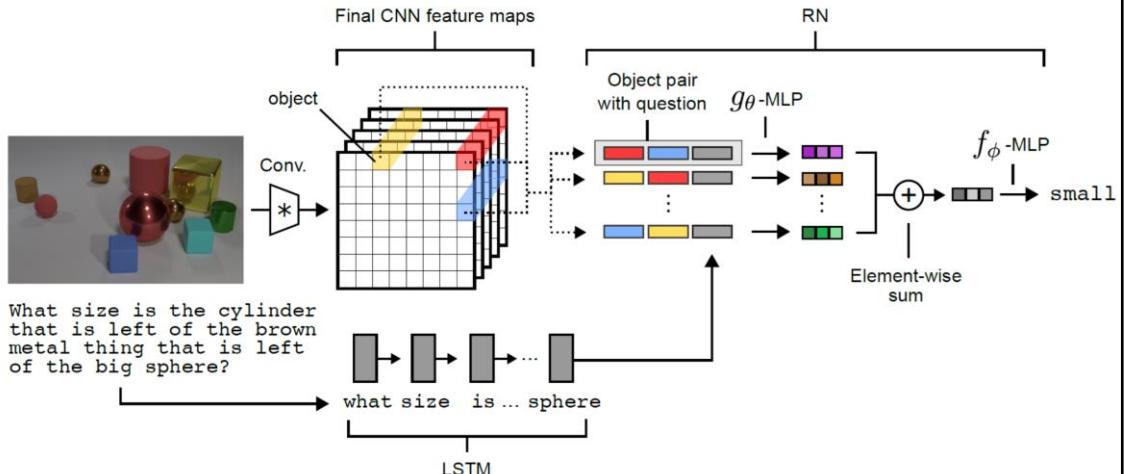
Failure cases. Many of these are difficult for humans too.

## Future work, applications

- Rich scene understanding in RL agents
- Social network modeling
- Abstract problem solving
- ” Relation Networks are a simple and powerful approach for learning to perform rich, structured reasoning in complex, real-world domains.”

## A more critical view

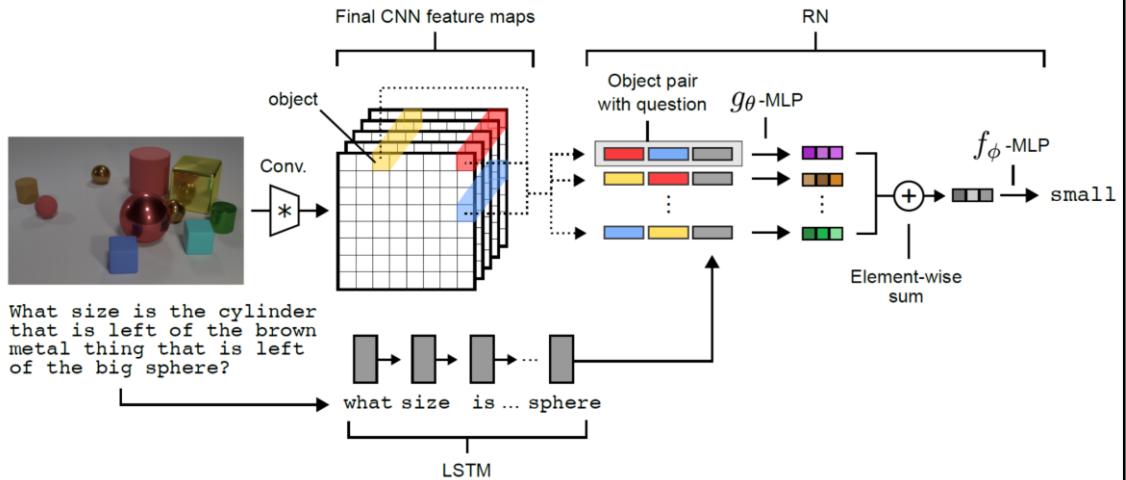
- The architecture exploits the weakness of the CLEVR benchmarks?
- Need a benchmark with large & complex objects, e.g., human crowds?



In particular, CLEVR images only have small objects such that a single CNN output pixel (a vector of values of all CNN feature maps at the same location) can reasonably represent the presence/absence of objects of different types, and the number of feature maps is hand-tuned to represent the number of object types in the dataset. The authors say that performance decreases if they increase the number of feature maps from 24, which probably causes the CNN to start distinguishing not only between objects but between variations within the same object (e.g., rotations), which can obviously cause problems for the RN or at least can need much more data for disambiguation.

Maybe a low-hanging fruit for a paper: create a synthetic visual benchmark like CLEVR but with 3d humanoids of different clothing, poses, gender, age – could be done easily with Unity and free 3d assets.

## Questions, discussion?



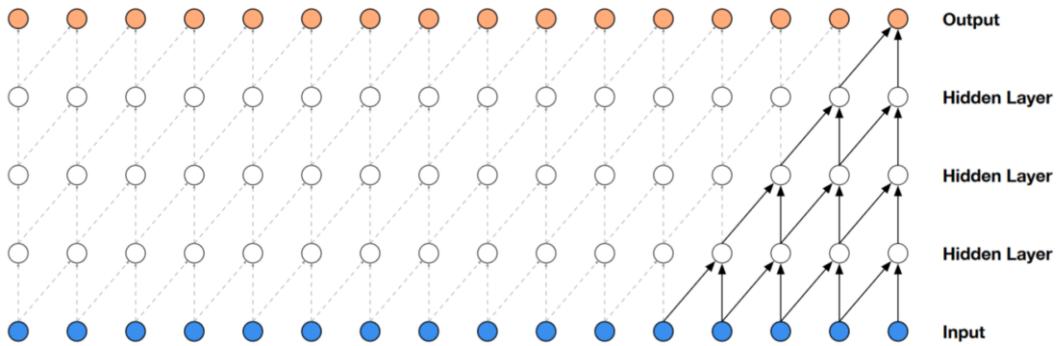
I was already asked how the RN can count. In my understanding, that would happen through the summation before the "f" network, provided that the "g" networks output the same vector for all object pairs that match the definition of pairs of objects to count, and some other vector for others. The "f" network can then learn to map the sum to the scalar answer. Counting single objects works the same way, as there are "g" networks for all pairs where the object is paired with itself ("equal" relation).

Relation networks summary end

## Alternative to recurrent networks: Dilated causal convolution

- Popularized by Deepmind Wavenet:

<https://deepmind.com/blog/wavenet-generative-model-raw-audio/>



<https://arxiv.org/pdf/1609.03499.pdf>

Show the animation from the DeepMind blog!

This is more stable to train than RNN:s, but on the other hand, requires more computing resources after training (unless substantial engineering effort invested).

## Audio synthesis

- Tacotron & Tacotron 2 build on WaveNet, Tacotron 2 used in Google's voice assistant

- Online tools for synthesizing speech:

<https://text-to-speech-demo.ng.bluemix.net/>

<https://acapela-box.com/AcaBox/index.php> (Try the "Will bad guy" voice)

<https://lyrebird.ai/> (text-to-speech mimicking any voice, just input a few phrases)

acapela-box does not yet appear to use deep learning, but it's probably just a matter of time.

The screenshot shows a user interface for text-to-speech configuration. At the top, there are three tabs: 'Text', 'SSML' (which is selected and highlighted in purple), and 'Voice Transformation SSML'. Below the tabs is a large text input area containing the following SSML code:

```
<p><s>Conscious of its spiritual and moral heritage <break time="300ms"/>, the Union is founded on the indivisible, universal values of <prosody rate="-15%">human dignity, freedom, equality and solidarity.</prosody> It is based on the principles of democracy and the rule of law <break time="500ms"/>. </s> <s> It places the individual at the heart of its activities, <prosody rate="+15%">by establishing the citizenship of the Union</prosody> and by creating an area of freedom, security and justice.</s></p>
```

At the bottom of the text input area, there is a horizontal scrollbar with arrows. To the right of the main content area is a vertical scroll bar. At the very bottom of the interface, there is a URL: <https://text-to-speech-demo.ng.bluemix.net/>.

Typically, these services provide some form of markup language that let's you finetune the results. With some tweaking, the results are usable for game prototypes, video voice overs etc.

## Sonantic obtains a financing of 2.3 million euros to bring artificial "human quality" voices to the games – Newsdio

By admin - March 3, 2020

477 0



It seems there's quite push to turn these technologies into products. I don't know whether Sonantic's tech is based on RNN:s or Wavenet, but I'm sure this is happening in a few years and will empower, e.g., indie game developers.

As a downside, this will deprive voiceover freelancers from whatever income they are now able to generate through services like voice123.com.



## Case study 2: OpenAI GPT-2

ABOUT PROGRESS RESOURCES BLOG



# Better Language Models and Their Implications

We've trained a large-scale unsupervised language model which generates coherent paragraphs of text, achieves state-of-the-art performance on many language modeling benchmarks, and performs rudimentary reading comprehension, machine translation, question answering, and summarization—all without task-specific training.

FEBRUARY 14, 2019  
21 MINUTE READ

SYSTEM PROMPT (HUMAN-WRITTEN)

*In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.*

MODEL COMPLETION (MACHINE-WRITTEN, 10 TRIES)

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.

Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them - they were so close they could touch their horns.

While examining these bizarre creatures the scientists discovered that the creatures also spoke some fairly regular English. Pérez stated, "We can see, for example, that they have a common 'language,' something like a dialect or dialectic."

## Simple but powerful

- The network is trained to predict text based on preceding text
- Can be primed with a small segment of text
- Applications: text generation, of course, but also natural language question answering
- Full model not available because of enormous misuse potential for fake news generation etc.
- A "small" 107M parameter pre-trained network available
- Results: <https://openai.com/blog/better-language-models/>

## Key success factors

- Transformer architecture (Vaswani et al. 2017), so called "multi-head attention"
- Clever dataset curation: Use Reddit karma as a proxy metric for relevance.
- The network was trained on all Reddit outgoing links with over 2 karma
- Again, collecting the right data is often more important than technical innovations

# Attention Is All You Need

Ashish Vaswani\*  
Google Brain  
avaswani@google.com

Noam Shazeer\*  
Google Brain  
noam@google.com

Niki Parmar\*  
Google Research  
nikip@google.com

Jakob Uszkoreit\*  
Google Research  
usz@google.com

Llion Jones\*  
Google Research  
llion@google.com

Aidan N. Gomez\* †  
University of Toronto  
aidan@cs.toronto.edu

Lukasz Kaiser\*  
Google Brain  
lukasz.kaiser@google.com

Illia Polosukhin\* ‡  
illia.polosukhin@gmail.com

## Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

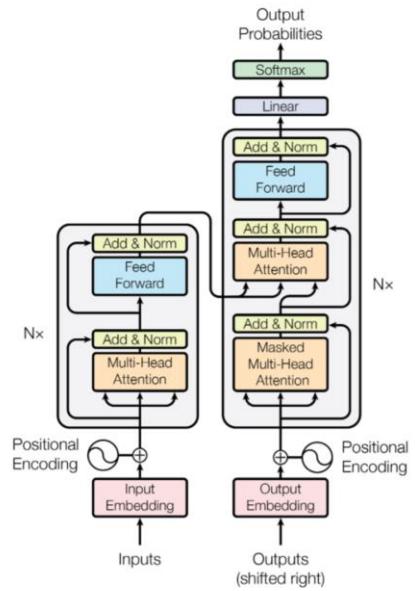
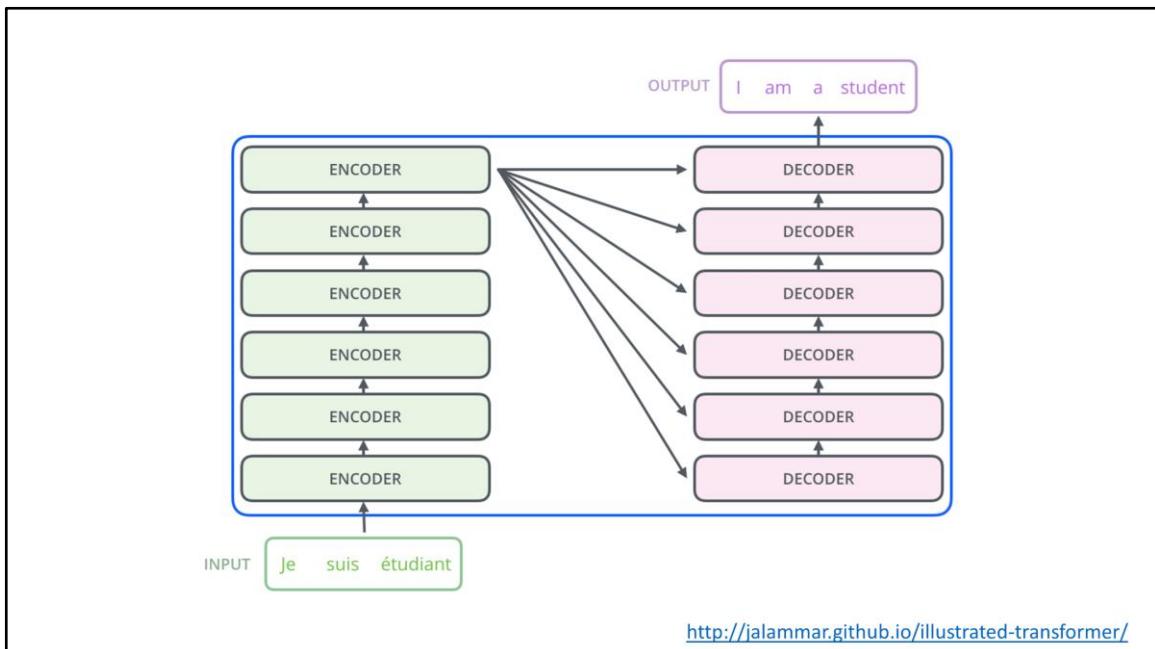


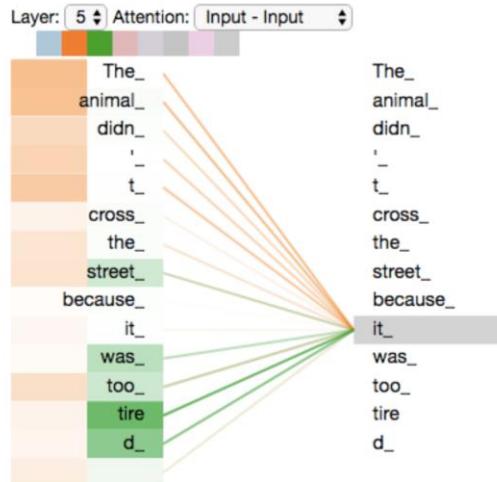
Figure 1: The Transformer - model architecture.

## The original transformer paper

<https://arxiv.org/pdf/1706.03762.pdf>



The original paper is not the easiest read. This project visualizes things more clearly.



As we encode the word "it", one attention head is focusing most on "the animal", while another is focusing on "tired" -- in a sense, the model's representation of the word "it" bakes in some of the representation of both "animal" and "tired".

<http://jalammar.github.io/illustrated-transformer/>

The different colors illustrate different attention heads



Grace Lindsay  
@neurograce

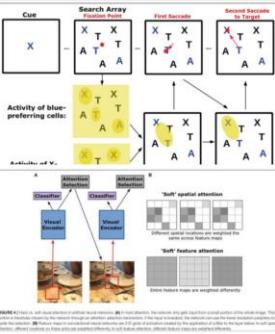
Have you heard the word "attention" thrown around in both neuroscience & machine learning? Have you wondered if/how its different uses relate to each other? My new review aims to summarize how this giant topic is studied & modeled across different domains! [frontiersin.org/articles/10.33...](https://frontiersin.org/articles/10.33...)

## Attention in Psychology, I and Machine Learning

Grace W. Lindsay\*

Gatsby Computational Neuroscience Unit, Sainsbury Wellcome Centre  
Kingdom

Attention is the important ability to flexibly control how information is processed. It has been studied in conjunction with many other topics including awareness, vigilance, saliency, executive control, and memory. It has recently been applied in several domains in machine learning, including the study of biological attention and its use as a tool to improve the performance of machine learning models. However, the relationship between attention and machine learning is not always clear. This review starts by providing an overview of the different ways attention is conceptualized in the neuroscience and psychology literature, and then provides examples of attention in machine learning, indicating their similarities and differences. Finally, the ways in which artificial attention can be used to improve the performance of machine learning models are discussed.



<https://twitter.com/neurograce/status/1250745465261436931>

For those who really want to understand attention.

## GPT-2 applications

- Poetry, song lyrics, food recipes, subreddit simulations...
- Try it yourself in the browser: <https://talktotransformer.com/>
- Very often, the results include garbage.
- Needs human curation, e.g., selecting the best of every 10 generations

With the right curation, the results can be super interesting.

(Take a brief pause, ask students to try it out, and paste the best curated results to the chat)

# GPT-2 NEURAL NETWORK POETRY

Demonstration tutorial of retraining OpenAI's GPT-2 (a text-generating Transformer neural network) on large poetry corpuses to generate high-quality English verse.

[statistics](#), [NN](#), [fiction](#), [shell](#), [GPT](#), [tutorial](#), [poetry](#)

3 March 2019–29 Oct 2019 · finished · certainty: likely · importance: 7

- 1 GPT-2-117M: Generating Poetry
- 2 Training GPT-2-117M To Generate Poetry
  - 2.1 Data: The Project Gutenberg Poetry Corpus
- 3 Training GPT-2-poetry
  - 3.1 GPT-2-poetry Samples
  - 3.2 Cleaning Project Gutenberg & Contemporary Poetry
- 4 Training GPT-2-poetry-prefix
  - 4.1 GPT-2-poetry-prefix Samples
    - 4.1.1 Training samples
    - 4.1.2 Unconditional samples
  - 4.2 GPT-2-poetry-prefix completions
    - 4.2.1 "Howl"
    - 4.2.2 "Ozymandias"
    - 4.2.3 *Essay on Criticism*
    - 4.2.4 8 Famous First Lines
      - 4.2.4.1 "Illyria" Lord Alfred Tennyson

In February 2019, following up on my ["2015–2016 text-generation experiments with char-RNNs"](#), I experiment with the cutting-edge Transformer NN architecture for language modeling & text generation. Using OpenAI's GPT-2-117M (117M) model pre-trained on a large Internet corpus and nshepperd's finetuning code, I retrain GPT-2-117M on a large (117MB) Project Gutenberg poetry corpus. I demonstrate how to train 2 variants: "GPT-2-poetry", trained on the poems as a continuous stream of text, and "GPT-2-poetry-prefix", with each line prefixed with the metadata of the PG book it came from. In May 2019, I trained the next-largest GPT-2, 345M, similarly, for a further quality boost in generated poems. In October 2019, I retrained 117M on a Project Gutenberg corpus with improved formatting, and combined it with a contemporary poem dataset based on ["Poetry Foundation"](#)'s website.

<https://www.gwern.net/GPT-2>



wint but AI (closed)

@drilgpt2archive

▼

the only time I flush the toilet is if I'm trying to kill something with it

3:59 AM · Dec 28, 2019 · [dril\\_gpt2](#)

---

187 Retweets 1K Likes

Twitter bot, human-curated GPT-2

https://theselyricsdonotexist.com Haku

# These Lyrics Do Not Exist

Lyrics generated using Artificial Intelligence

This website generates completely original lyrics for various topics, uses state of the art AI to generate an original chorus and original verses  
You can choose the AI songwriter lyrics topic, lyrics genre and lyric mood

## Generate Your Own Lyrics

Type a song topic word and then press the "Generate My Lyrics" button  
example Song Topics: [love](#), [party](#), [dance](#), [happy](#), or human names such as [peter](#), [jennifer](#), [bella](#), [alex](#),  
or you can even generate songs by emoji or etc...

Song Topic:

Lyrics Genre:

Lyric Mood:

Generate My Lyrics

[-] **itsmybirthday19** [S] 76 points 6 months ago  
Input: 🍺 (beer emoji)

Output:

Verse 1  
Dad I had to have a 🍺  
To help me get through another day  
When I think of the lady I love  
Time to drink that 🍺

Pre-Chorus  
They were drinking 🍺 outside of my father's house  
Let's take our time with this 🍺

Chorus  
Cold 🍺 on the floor  
And I can see you now  
To change those words you feed me with lies  
I chug 🍺 🍺 🍺 🍺

Verse 2  
I've come here again to drink my 🍺 again.  
I've come here again to drink my 🍺 again.  
Oh I'm goin' right on! don't you see?  
Just 'cause I look just fine

Pre-Chorus  
One more 🍺 and I'll be in no state to do the things I think I need to  
A place to shake my 🍺

<https://www.gwern.net/docs/www/old.reddit.com/14b0aa1719ce16b97ddd47ba7e3020485fd64292.html>

[-] localkhaleesi 11 points 6 months ago

Input: fuck

Verse 1

What the f\*\*k what the f\*\*k what the f\*\*k?  
The same they claim when problems pop  
Talk mess up talk mess up talk mess up  
The same they claim when problems pop

Pre-Chorus

I see the same thing all over the f\*\*king world every single day  
I see the same thing in my women f\*\*k the greed the fake

Chorus

Get the f\*\*k outta here woah I'm on the up  
See I don't know about you f\*\*k boys  
Tell me you're not scared f\*\*k that  
I really don't give a f\*\*k about you girl

Verse 2

Turn it up and f\*\*k the sun  
Cause I know that you count on me  
And that's where you belong  
Make your lady wanna f\*\*k me

Pre-Chorus

Don't f\*\*k with me no more oh I'm so cool  
Don't f\*\*k with me no more

<https://www.gwern.net/docs/www/old.reddit.com/14b0aa1719ce16b97ddd47ba7e3020485fd64292.html>

You are Zorro, a wizard living in the kingdom of Larion. You have a staff and a spellbook. You finish your long journey and finally arrive at the ruin you've been looking for. You look around and see that it's just as you remember. The ruins are all that remain of the once great city of Larion. The only thing that looks different is the fact that there are no more buildings to be seen

> invent the internet

You think about how you can use this opportunity to make the world a better place. You decide to go back to the ruins and try to figure out what happened to the people who lived here before you arrived.

The next day you return to the ruins with your new found knowledge

> invent the internet

You begin by creating a computer program which will allow you to communicate with anyone on the net. It takes some time but eventually you get it working. You then start sending messages to other wizards who live in the area

AI Dungeon: <https://www.rockpapershotgun.com/2019/12/09/this-ai-text-adventure-generator-lets-you-do-anything-you-want/>

Also using GPT-2. Possibly the best use of deep learning for procedural game content I've seen. Simple to implement (just finetune an existing model with custom data, add a few lines of code for interaction), but with huge impact, granting players unprecedented freedom of expression.

<https://aiweirdness.com/post/190569291992/ai-recipes-are-bad-and-a-proposal-for-making-them>



Pictured above is an abomination in the making, a lesson in why humans should never trust what a neural net says just because it's based on math. It's a neural net generated brownie recipe called [Chocolate Baked and Serves](#), and its distinguishing feature is the CUP OF HORSERADISH it contains. It was so bad that my eyes watered as I removed it from the oven.

## Summary

- Both CNNs and RNNs can be used for sequence modeling
- LSTM and GRU networks can have long memory with small computing cost
- Convolutional networks for very long sequences, e.g., the WaveNet: dilated causal convolutions
- CNNs and RNNs can be combined in many ways, e.g., image captioning, Relational Networks
- State of the art in text generation: Transformer networks (e.g., GPT-2)
- Surprisingly good results in audio generation: SampleRNN
- Applications: text generation, image captioning, audio synthesis, animation synthesis