

Title: Attention is all you get

Speakers: Paul Ginsparg

Collection: Machine Learning for Quantum Design

Date: July 11, 2019 - 10:45 AM

URL: <http://pirsa.org/19070013>

Abstract: For the past decade, there has been a new major architectural fad in deep learning every year or two.

One such fad for the past two years has been the transformer model, an implementation of the attention method which has superseded RNNs in most sequence learning applications. I'll give an overview of the model, with some discussion of non-physics applications, and intimate some possibilities for physics.

# Attention is all you get

**Paul Ginsparg**

**Physics and InfoSci, Cornell University**

For the past decade, there has been a new major architectural fad in deep learning every year or two. One such fad for the past two years has been the transformer model, an implementation of the attention method which has superseded RNNs in most sequence learning applications. I'll give an overview of the model, with some discussion of non-physics applications, and intimate some possibilities for physics.

“Machine Learning for Quantum Design”, Perimeter Institute, 11 Jul 2019

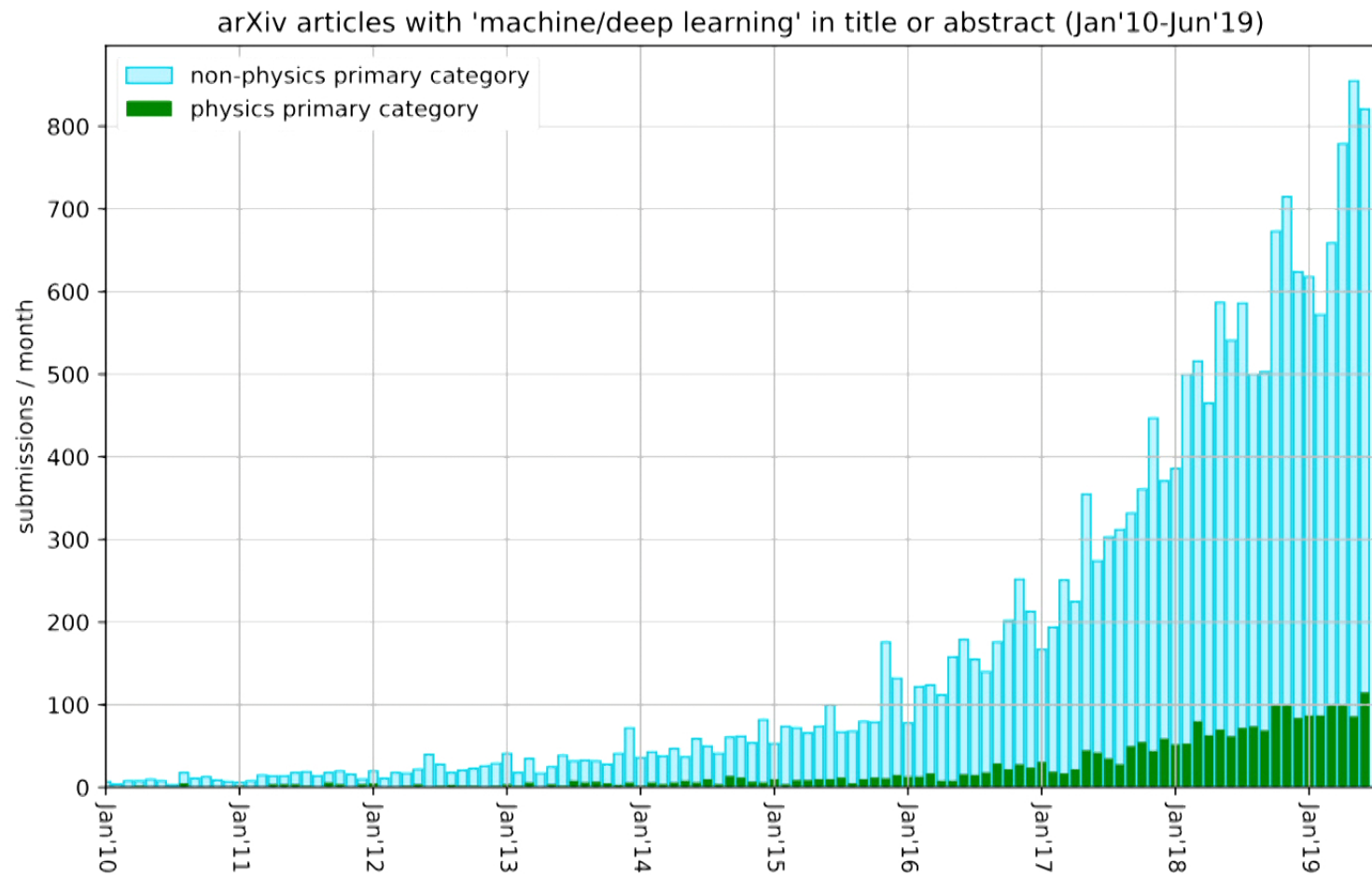
## Attention is all you get

**Paul Ginsparg**

**Physics and InfoSci, Cornell University**

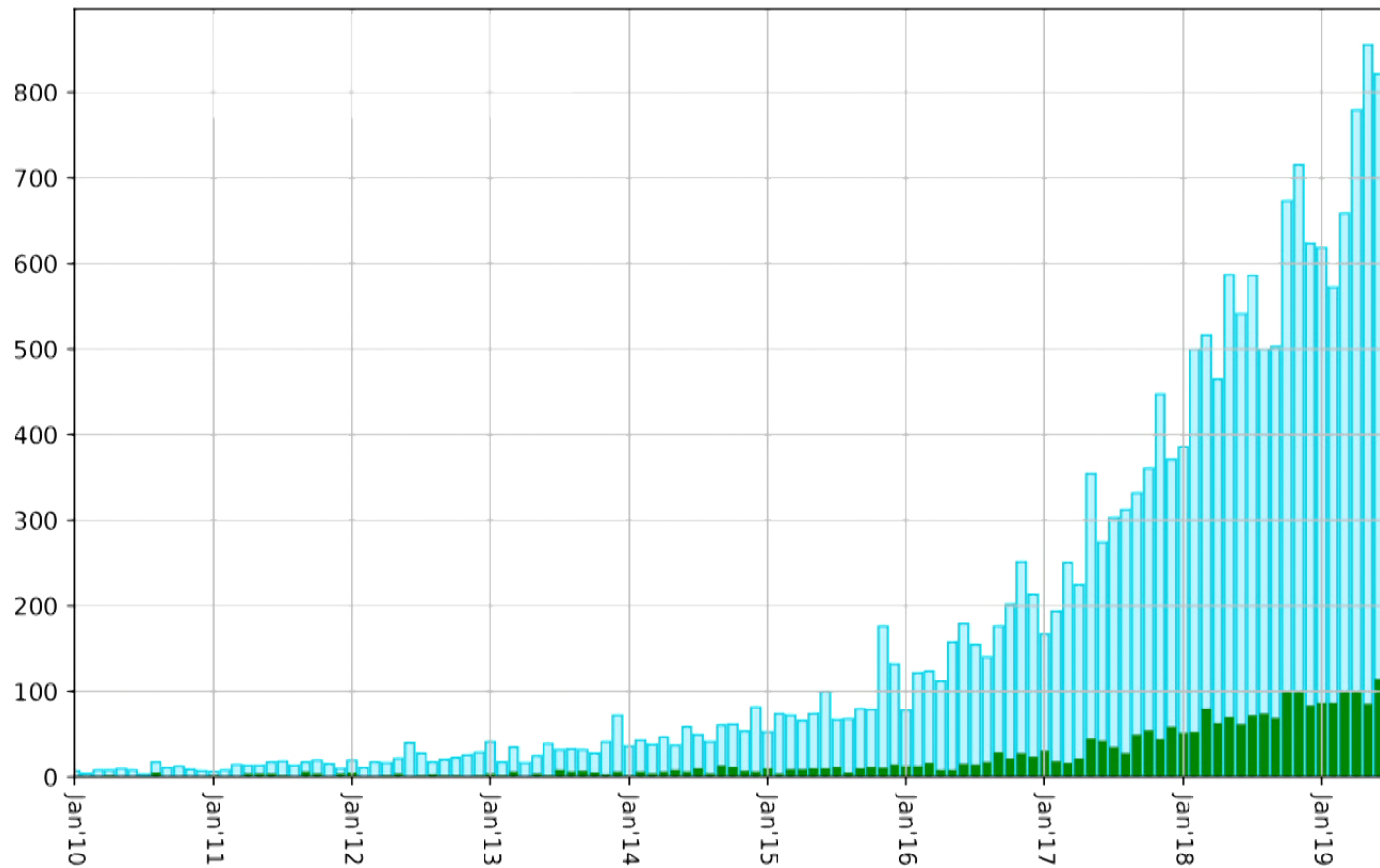
For the past decade, there has been a new major architectural fad in deep learning every year or two. One such fad for the past two years has been the transformer model, an implementation of the attention method which has superseded RNNs in most sequence learning applications. I'll give an overview of the model, with some discussion of non-physics applications, and intimate some possibilities for physics.

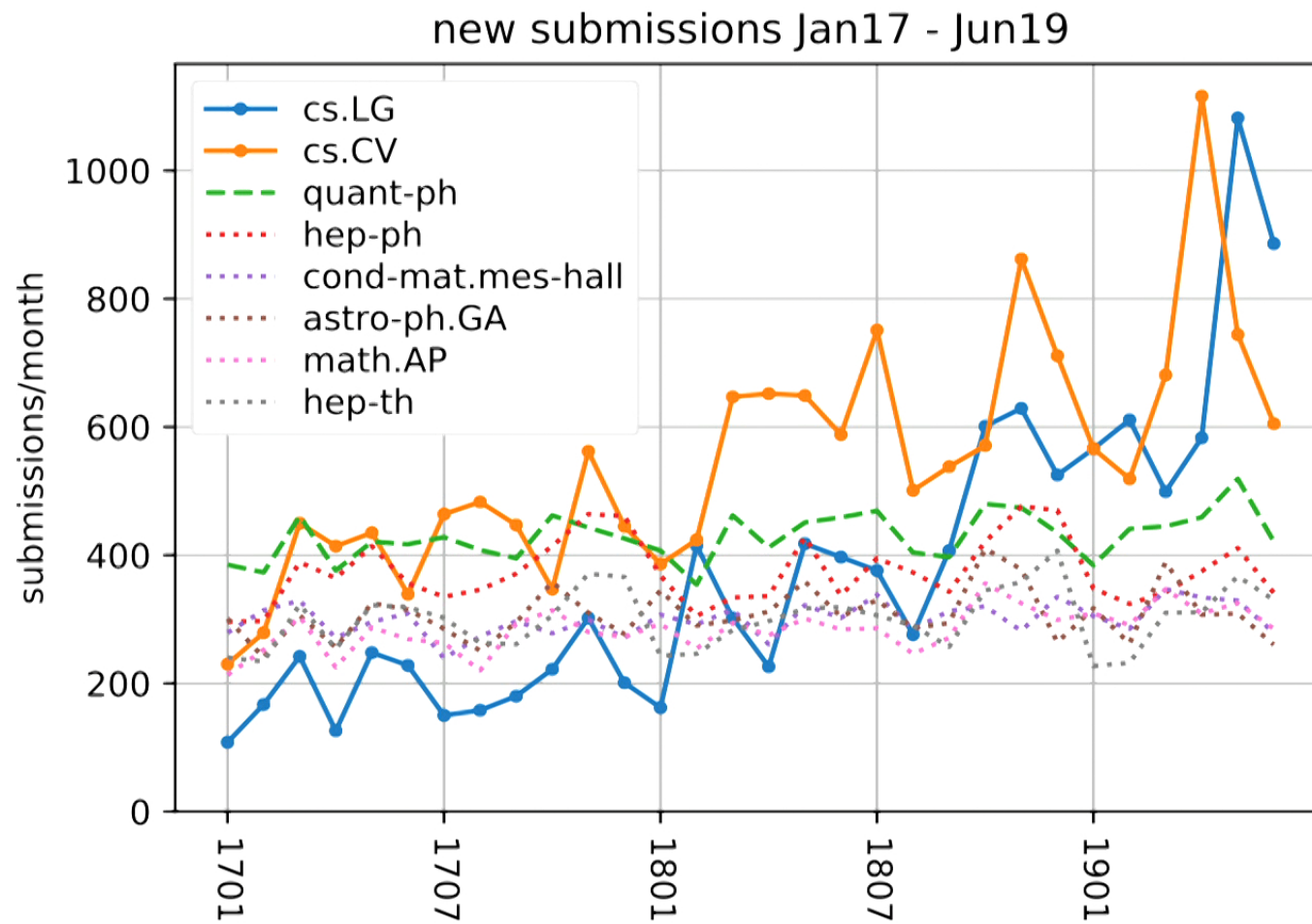
“Machine Learning for Quantum Design”, Perimeter Institute, 11 Jul 2019





## Cornell Undergrad majors in cs/info





[from Juan Carrasquilla]

<https://openai.com/blog/better-language-models/> (Feb 2019)

GPT-2 = Generative Pre-training Transformer

## EXAMPLE OF A LANGUAGE MODEL (TRANSFORMER)

SYSTEM PROMPT (HUMAN-WRITTEN)	<i>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.</i>
MODEL COMPLETION (MACHINE-WRITTEN, 10 TRIES)	<p>The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science.</p> <p>Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>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."</p> <p>Dr. Pérez believes that the unicorns may have originated in Argentina, where the animals were believed to be descendants of a lost race of people who lived</p>

← Input

← Output

generating samples from a variety of inputs —> close to human quality and long-range coherence over a **page or more of text**

↓  
Strong correlation (power law)

$$P(\text{output text} | \text{input text})$$

[from Juan Carrasquilla]

## LANGUAGE TRANSLATION

---

► Language translation  $P(\text{English}|\text{Spanish})$

### *Input*

Utilizando la base de datos Epistemonikos, la cual es mantenida mediante búsquedas realizadas en 30 bases de datos, identificamos seis revisiones sistemáticas que en conjunto incluyen 36 estudios aleatorizados pertinentes a la pregunta.

### *Output*

Using the Epistemonikos database, which is maintained through searches in 30 databases, we identified six systematic reviews that altogether include 36 randomized studies relevant to the question.

Neural Machine Translation with the Transformer and Multi-Source Romance Languages for the Biomedical WMT 2018 task.  
Brian Tubay and Marta R. Costa-jussà (2018) <https://www.aclweb.org/anthology/W18-6449>

Abstract begins:

The Transformer architecture has become the state-of-the-art in Machine Translation. This model, which relies on attention-based mechanisms, has outperformed previous neural machine translation architectures in several tasks. ...

Computer Science > Computation and Language

## Attention Is All You Need

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin

(Submitted on 12 Jun 2017 (v1), last revised 6 Dec 2017 (this version, v5))

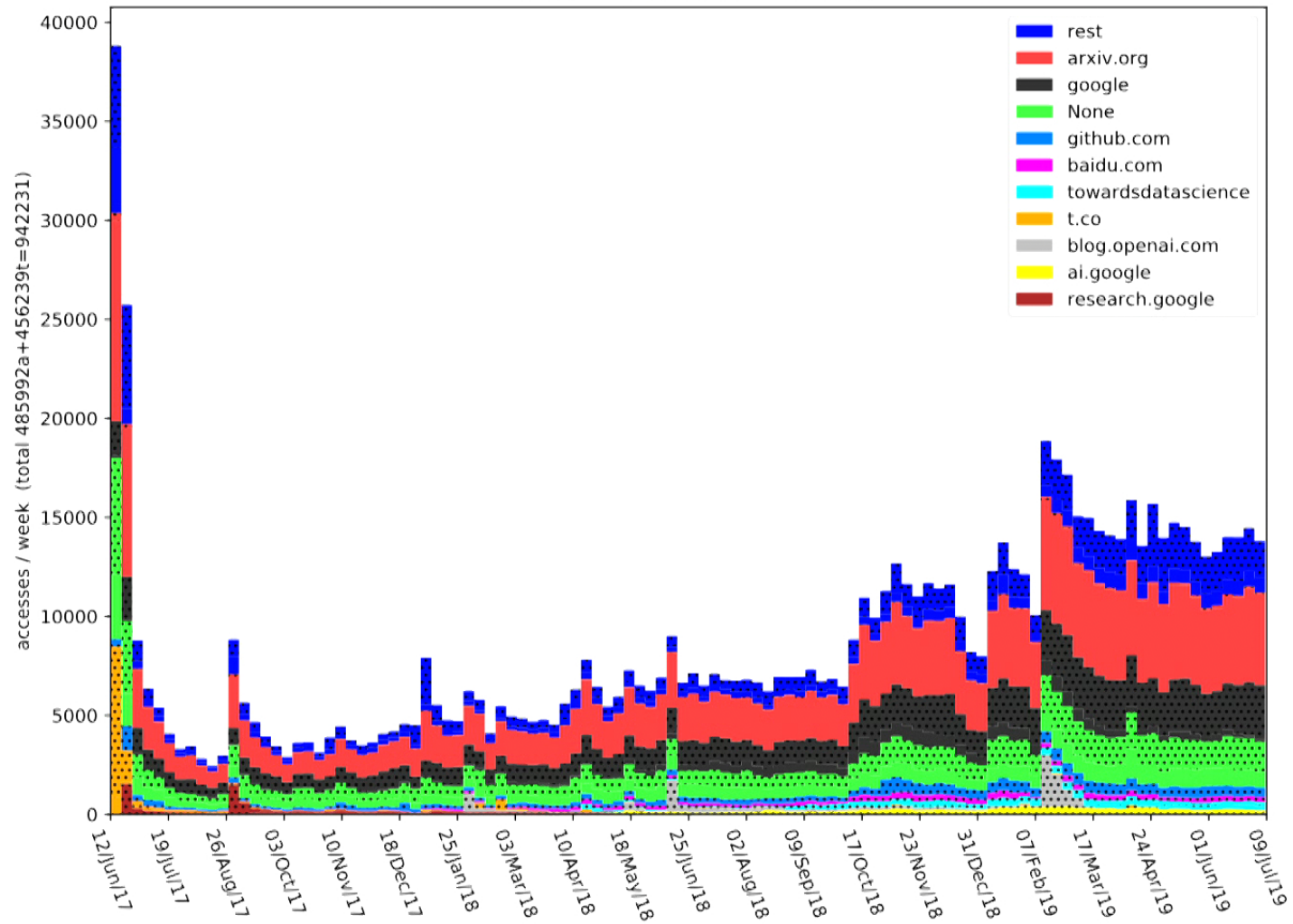
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks in an encoder-decoder configuration. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, 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.

Comments: 15 pages, 5 figures

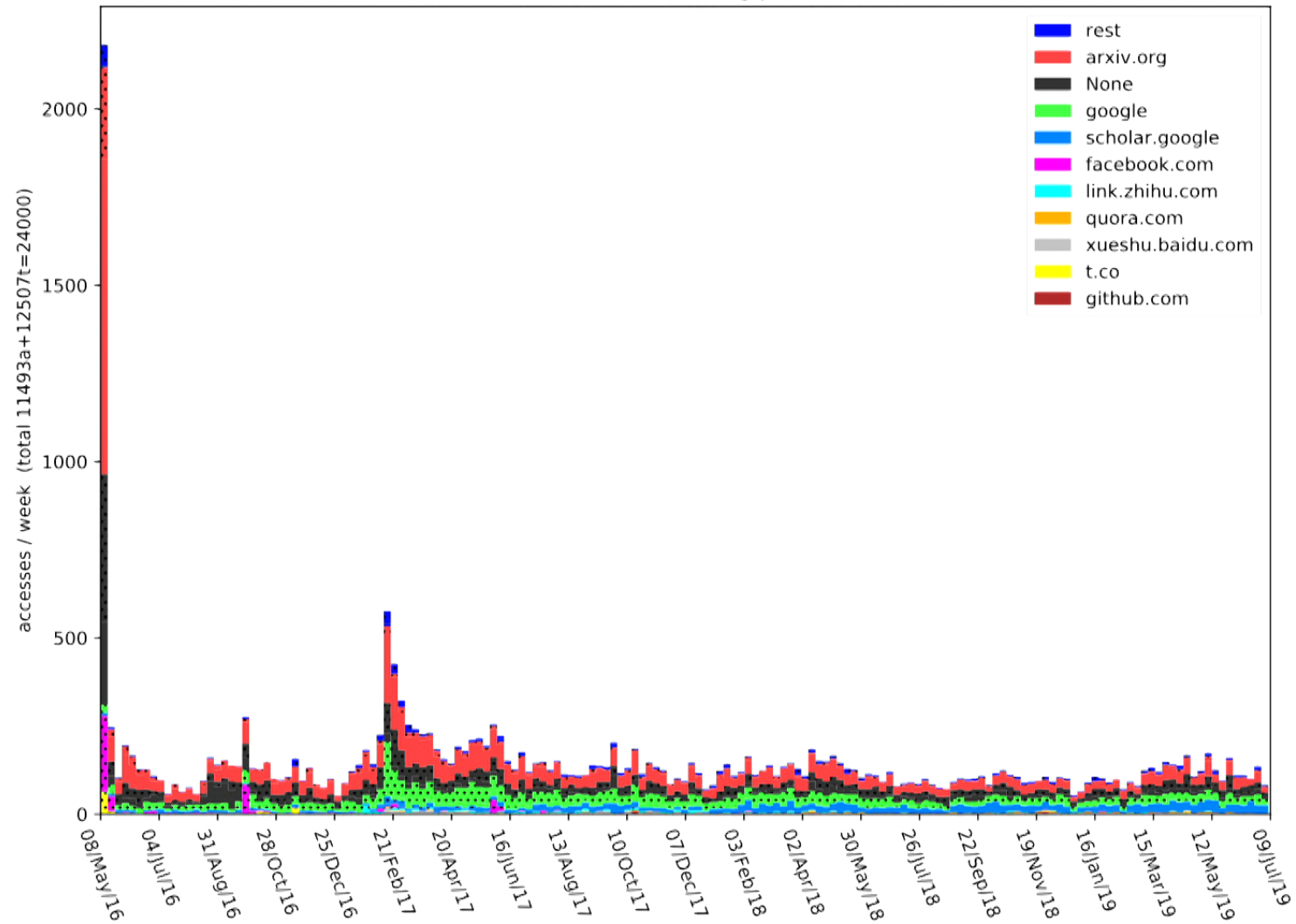
Subjects: **Computation and Language (cs.CL)**; Machine Learning (cs.LG)

(and > 2k citations)

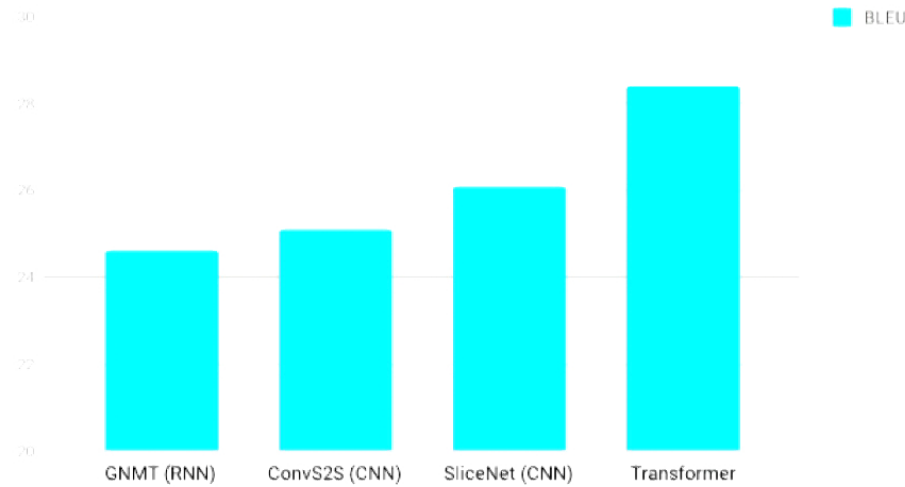
1706.03762 Attention Is All You Need



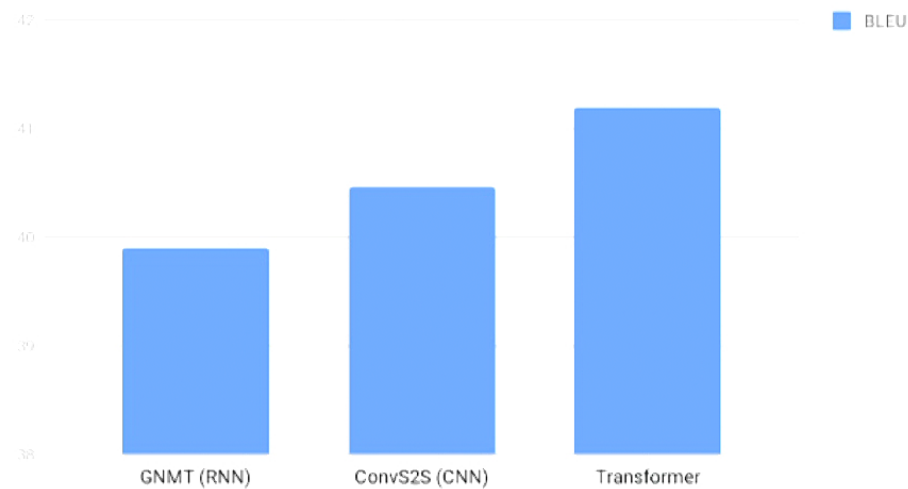
1605.01735 Machine learning phases of matter



English German Translation quality



English French Translation Quality



12 hours on 8 P100 GPUs,  
each many millions of  
sentence pairs.

Transfer Learning:  
also good at other tasks  
(reading comp, Q/A;  
summarization,  
sentence reps)

BLEU = Bilingual Evaluation Understudy

What is a language model?

## “Mark V Shaney”

```
def make_trigrams(filename):  
  
    with open(filename) as f: words = f.read().split()  
  
    trigrams = defaultdict(list)  
  
    bigram=tuple(words[:2])  
  
    for w in words[2:] + words[:2]:  
        #keys of trigram dict are tuples, values are lists  
        trigrams[bigram].append(w)  
        bigram=(bigram[1],w)  
  
    return trigrams
```

```
trigrams_ml[('based','on')] → [ ('the', 26),  
                                ('a', 23),  
                                ('quantum', 9),  
                                ('machine', 4),  
                                ('an', 3),  
                                ('classical', 3),  
                                ('artificial', 3),  
                                ('deep', 3),  
                                ('estimating', 3),  
                                ('Bayesian', 2),  
                                . . .]
```

```
trigrams_ml[('on','a')] → [ ('quantum', 20),  
                             ('single', 4),  
                             ('classical', 3),  
                             ('small', 3),  
                             ('diverse', 2),  
                             ('subset', 2),  
                             ('simple', 2),  
                             ('hidden', 2),  
                             ('large', 2),  
                             ('restricted', 2),  
                             . . .]
```

```
def random_text(trigrams, startwords, num_words=100):
```

```
    current_pair = random.choice(startwords)
    random_text = list(current_pair)
```

```
    # continue past num_words until ends in .
```

```
    while len(random_text) < num_words:
```

```
        next = random.choice(trigrams[current_pair])
```

```
        random_text.append(next)
```

```
        current_pair = (current_pair[1], next)
```

```
        # avoid long loops if too few periods
```

```
        # in training text
```

```
    return ''.join(random_text)
```

Chrome File Edit View History Bookmarks People Window Help

tmp/ x trigram\_gen x wvec x Deconstructing BERT: Distilling x +

localhost:8888/notebooks/tmp/trigram\_gen.ipynb

Apps MathJax zzz zzz ac ao aq bn fc h hr meta nbn sp trn uc admin pk tur S2 twspeed lp

jupyter trigram\_gen (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

return textwrap.fill(' '.join(random\_text))

```
In [4]: trigrams_ml, startwords_ml = make_trigrams('ml.txt.gz')
```

```
In [ ]: #most common bigram keys
sorted([(bi,len(trigrams_ml[bi])) for bi in trigrams_ml],key=lambda x: x[1])
```

```
In [ ]: Counter(trigrams_ml[('based','on')]).most_common(10)
```

```
In [ ]: Counter(trigrams_ml[('on','a')]).most_common(10)
```

```
In [ ]: Counter(startwords_ml).most_common(10) #starts of sentences
```

```
In [12]: print (random_text(trigrams_ml,startwords_ml))
```

Quantum artificial intelligence research. Our results open the route to accelerate training is a formally exact description of a given problem assisted by machine learning. We demonstrate both theoretically and numerically -- with a classical computer uses this information to allow for effective feature extraction by dimension reduction technique that groups "similar" data points and dimensionality. One of the Spherical Bessel descriptors satisfies all three main branches of the model construction from experimental data collection), avoiding the need for so-called oracularized variants of the expectation values of these restrictions is. Our simulations show an important ingredient to account for the Chimera topology.

In [ ]:

Chrome File Edit View History Bookmarks People Window Help

tmp/ x trigram\_gen x wvec x Deconstructing BERT: Distilling x +

localhost:8888/notebooks/tmp/trigram\_gen.ipynb

Apps MathJax zzz zzz ac ao aq bm fc h hr meta nbn sp trn uc admin pk tur S2 twspeed lp

jupyter trigram\_gen (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Code

```
return textwrap.fill(' '.join(random_text))
```

```
In [4]: trigrams_ml, startwords_ml = make_trigrams('ml.txt.gz')
```

```
In [ ]: #most common bigram keys
sorted([(bi, len(trigrams_ml[bi])) for bi in trigrams_ml.keys()])
```

```
In [ ]: Counter(trigrams_ml)
```

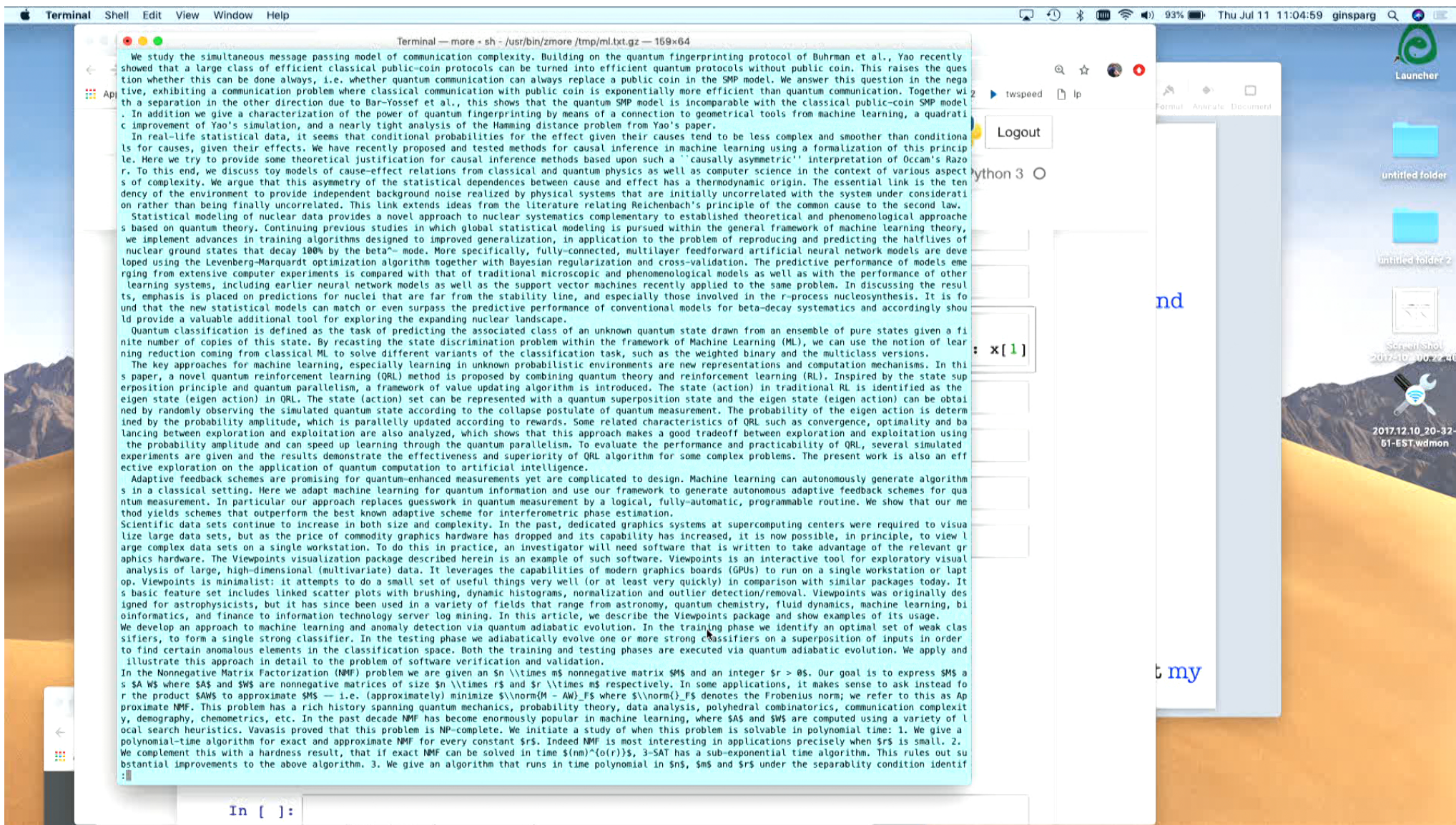
```
In [ ]: Counter(trigrams_ml)
```

```
In [ ]: Counter(startwords_ml).most_common(10) #starts of sentences
```

```
In [12]: print (random_text(trigrams_ml, startwords_ml))
```

Quantum artificial intelligence research. Our results open the route to accelerate training is a formally exact description of a given problem assisted by machine learning. We demonstrate both theoretically and numerically -- with a classical computer uses this information to allow for effective feature extraction by dimension reduction technique that groups "similar" data points and dimensionality. One of the Spherical Bessel descriptors satisfies all three main branches of the model construction from experimental data collection), avoiding the need for so-called oracularized variants of the expectation values of these restrictions is. Our simulations show an important ingredient to account for the Chimera topology.

In [ ]:



Chrome File Edit View History Bookmarks People Window Help

localhost:8888/notebooks/tmp/trigram\_gen.ipynb

jupyter trigram\_gen (unsaved changes) Logout Python 3

File Edit View Insert Cell Kernel Widgets Help

Code

```
return textwrap.fill(' '.join(random_text))
```

In [4]: trigrams\_ml, startwords\_ml = make\_trigrams('ml.txt.gz')

In [5]: m keys  
igrams\_ml[bi])) for bi in trigrams\_ml, key=lambda x: x[1], reverse=True)[:20]

Out[5]: [ (('of', 'the'), 700),  
 (('machine', 'learning'), 521),  
 (('in', 'the'), 315),  
 (('can', 'be'), 305),  
 (('of', 'quantum'), 237),  
 (('to', 'the'), 226),  
 (('of', 'a'), 218),  
 (('a', 'quantum'), 184),  
 (('for', 'the'), 182),  
 (('that', 'the'), 158),  
 (('the', 'quantum'), 155),  
 (('and', 'the'), 155),  
 (('In', 'this'), 150),  
 (('on', 'the'), 149),  
 (('show', 'that'), 149),  
 (('based', 'on'), 139),  
 (('number', 'of'), 136),  
 (('in', 'a'), 129),  
 (('neural', 'network'), 112),  
 (('such', 'as'), 109)]

In [ ]: Counter(trigrams\_ml[('based', 'on')]).most\_common(10)

Chrome File Edit View History Bookmarks People Window Help

localhost:8888/notebooks/tmp/trigram\_gen.ipynb

jupyter trigram\_gen (unsaved changes) Logout Python 3

File Edit View Insert Cell Kernel Widgets Help

Code

```
In [ ]:
```

```
In [ ]:
```

```
In [14]: print (random_text(trigrams_ml,startwords_ml))
```

We use HIP-NN, a neural network is proposed to help optimize the underlying objective function. Our quantum circuit defines a building block, the "quantum neuron", that can use a quantum algorithm for preparing states that decay 100% by the algorithm. We explain that a stochastic subgradient descent method that we can predict the ground state without any manual feature extraction. The performance of our approach for the tensor train decomposition and the standard AdaBoost to the DNN successfully learns differences in the efficiency of their output. This work can readily be extended to the number and type of the differential geometry and the logarithmic negativity.

```
In [17]: print (random_text(trigrams_ml,startwords_ml))
```

We provide a high performance in a common principled framework. We built on this analysis. We then move to describe the quantum GP algorithm is often extremely complicated and large, thus classical learning agents. Finally, works exploring the space and the Baryonic Oscillation Spectroscopic Survey (BOSS, data release 5 (DR5). Testing on other random instances from \$20\$ to \$28\$ bits continues to show enhanced performance and nearly equal ranking performance using the massive spatial multiplexing technique, to effectively boost the computational cost scales differently with electron number. Quantum

Chrome File Edit View History Bookmarks People Window Help

localhost:8888/notebooks/tmp/trigram\_gen.ipynb

jupyter trigram\_gen (unsaved changes) Logout Python 3

File Edit View Insert Cell Kernel Widgets Help

Code

built on this analysis. we then move to describe the quantum algorithm is often extremely complicated and large, thus classical learning agents. Finally, works exploring the space and the Baryonic Oscillation Spectroscopic Survey (BOSS, data release 5 (DR5). Testing on other random instances from \$20\$ to \$28\$ bits continues to show enhanced performance and nearly equal ranking performance using the massive spatial multiplexing technique, to effectively boost the computational cost scales differently with electron number. Quantum mechanics fundamentally forbids deterministic discrimination of quantum many-body systems. Quantum computing offers the potential for machine learning, generative models such as embedded and low-power GPUs, 2) it can combine multiple operations in half-precision floating-point FP16 saving bandwidth, time and  $\epsilon$  is the precision over classical hardware.

In [16]: `print (random_text(trigrams_ml,startwords_ml))`

This implies that BQP/qpoly is contained in the bipartite graph. Our model can be efficiently implemented on commercially-available quantum annealing machines produced by quantum measurement. The complexity of these examples turned into the dynamical behavior of the approach. Casimir physics covers a wealth of data by locally transforming the data such as the Higgs field is present. We show that the same family achieve better accuracy with significantly fewer iterations of generative training can be solved in time  $O(m \cdot \log n, 1/\epsilon)$  given access to measurements is limited. We also discuss deep-learning in finance, and suggestions to improve the effectiveness of the path for further quantum information theory.

In [ ]:

```
def random_text(trigrams, startwords, num_words=100):
```

```
    current_pair = random.choice(startwords)
```

```
    random_text = list(current_pair)
```

(We, both)

```
    # continue past num_words until ends in .
```

```
    while len(random_text) < num_words:
```

```
        next = random.choice(trigrams[current_pair])
```

```
        random_text.append(next)
```

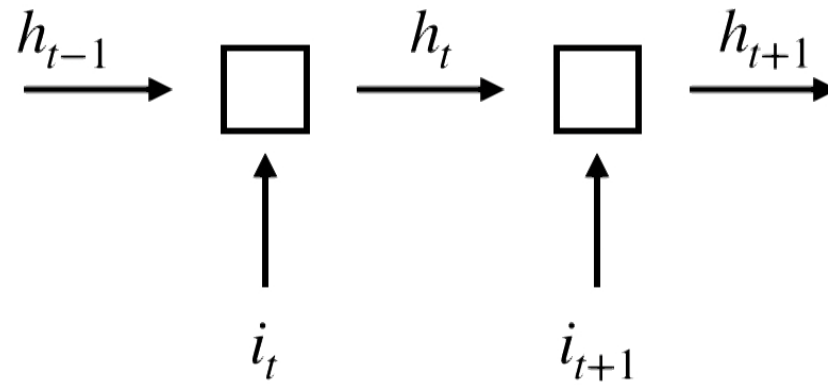
```
        current_pair = (current_pair[1], next)
```

```
        # avoid long loops if too few periods
```

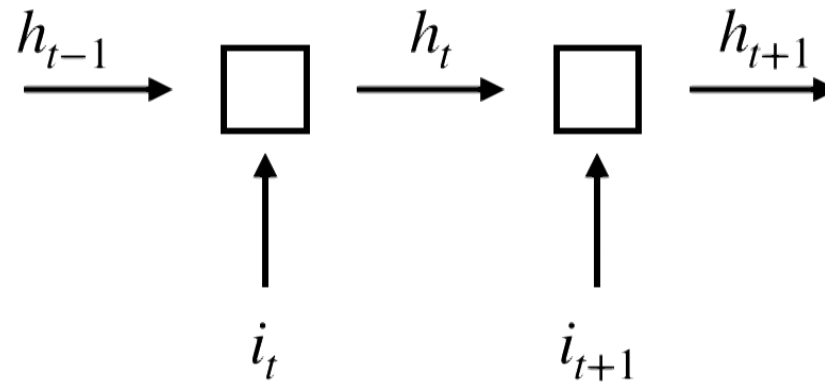
```
        # in training text
```

```
    return ''.join(random_text)
```

RNN = “recursive neural network”



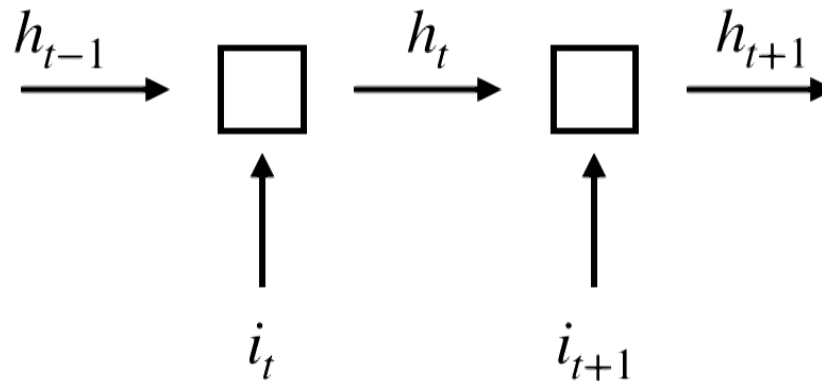
RNN = “recursive neural network”



*The animal didn't cross the street because it was too tired.*  
*L'animal n'a pas traversé la rue parce qu'il était trop fatigué.*

*The animal didn't cross the street because it was too wide.*  
*L'animal n'a pas traversé la rue parce qu'elle était trop large.*

RNN = “recursive neural network”

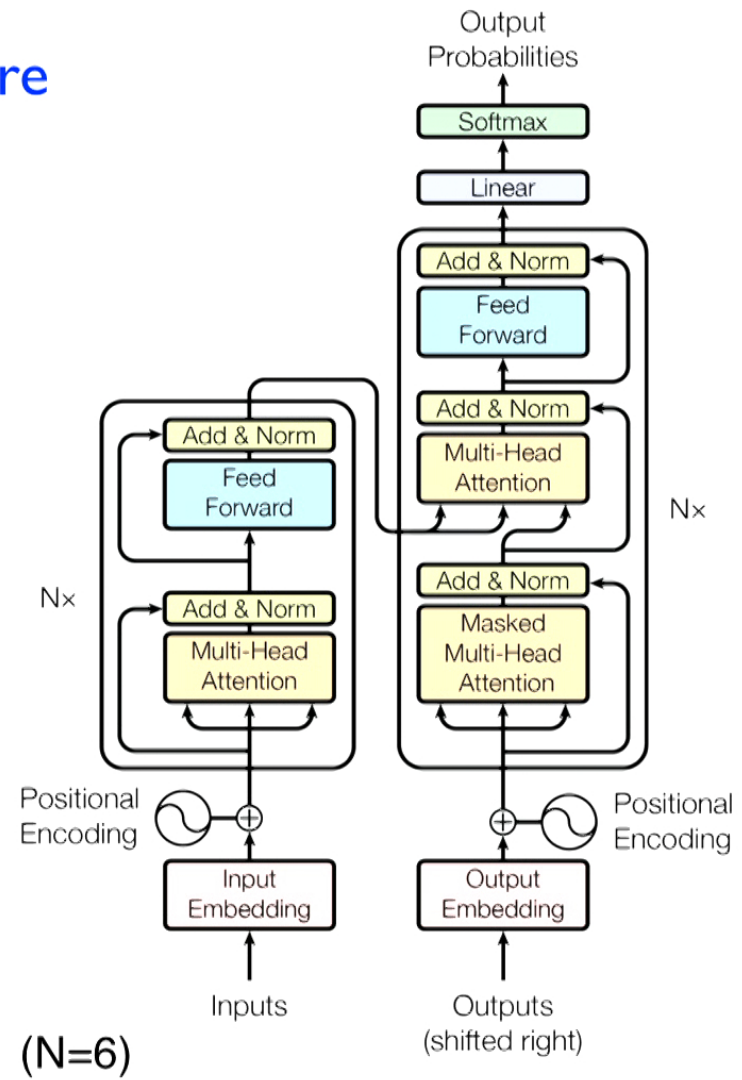


*The animal didn't cross the street because it was too tired.*  
*L'animal n'a pas traversé la rue parce qu'il était trop fatigué.*

*The animal didn't cross the street because it was too wide.*  
*L'animal n'a pas traversé la rue parce qu'elle était trop large.*

difficult to model long-range dependencies, difficult to parallelize, ...

# Transformer Architecture



# Attention

map query Q and set of key/value pairs K,V to an output:

$q_i$  = query vectors for words in sentence

$k_j, v_j$  = key/value pairs

(each is a  $d_k$  dimensional vector,  $i$  runs over words in sentence)

Attention of word  $i$  on word  $j$  is governed by the dot product  $q_i \cdot k_j$ , for all words  $j$ .

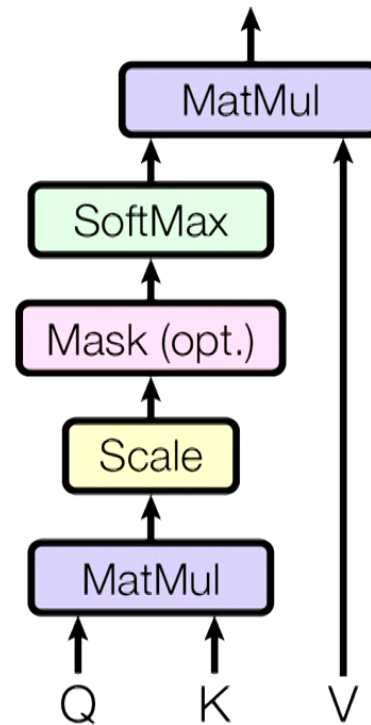
Specifically it is given by  $\text{softmax}\left(\frac{q_i \cdot k_j}{\sqrt{d_k}}\right)$

where  $\text{softmax} : \{x_i\} \rightarrow \left\{ \frac{\exp(x_i)}{\sum_j \exp(x_j)} \right\}$

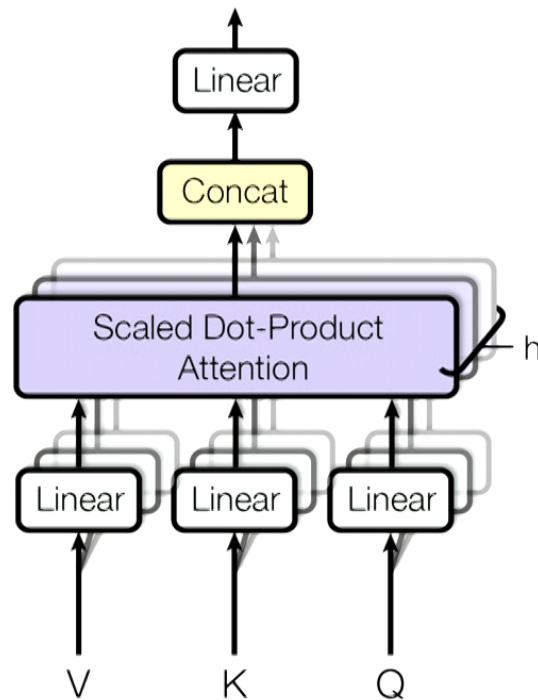
end result is  $v_i = \sum_j \text{softmax}\left(\frac{q_i \cdot k_j}{\sqrt{d_k}}\right) v_j$

In practice, stack them into matrices

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$



## Multi-headed Attention



$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O$$

$$W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}, \quad W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}, \quad W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v},$$

$$W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$$

$h = 8$  parallel attention layers, or heads.

For each of these, use  $d_k = d_v = d_{\text{model}}/h = 64$

Man bites dog

# Man bites dog

Positional Encoding:

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d_{\text{model}}})$$

(learn to attend by relative positions)

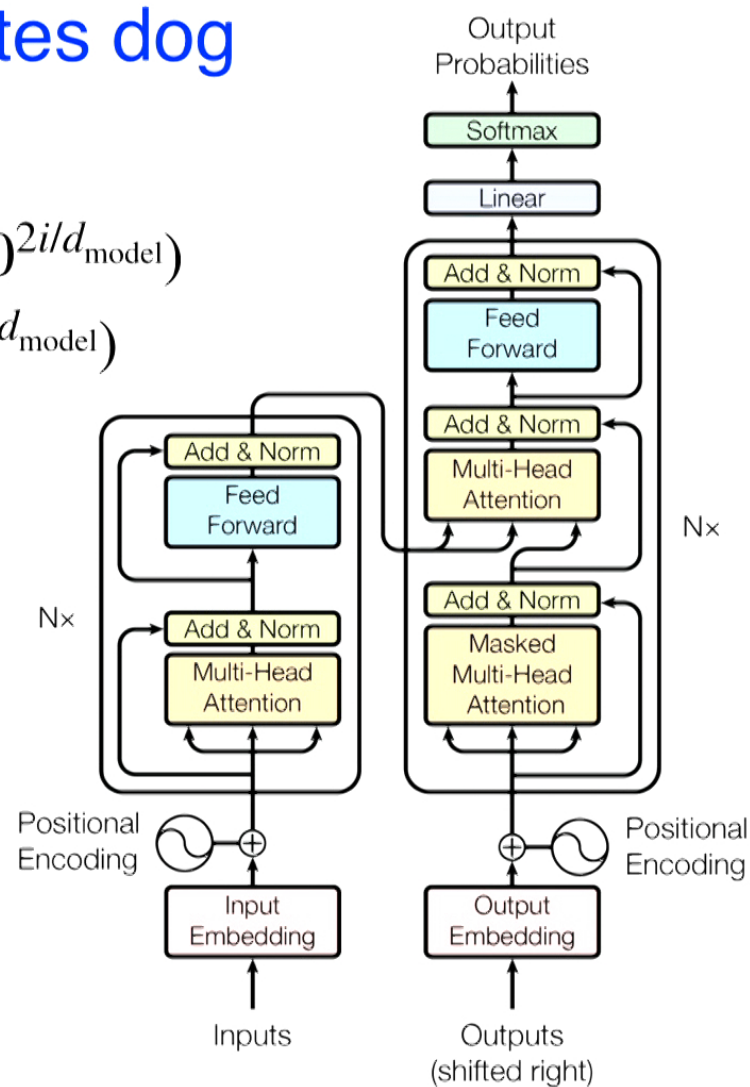
# Man bites dog

Positional Encoding:

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d_{\text{model}}})$$

(learn to attend by relative positions)



# Man bites dog

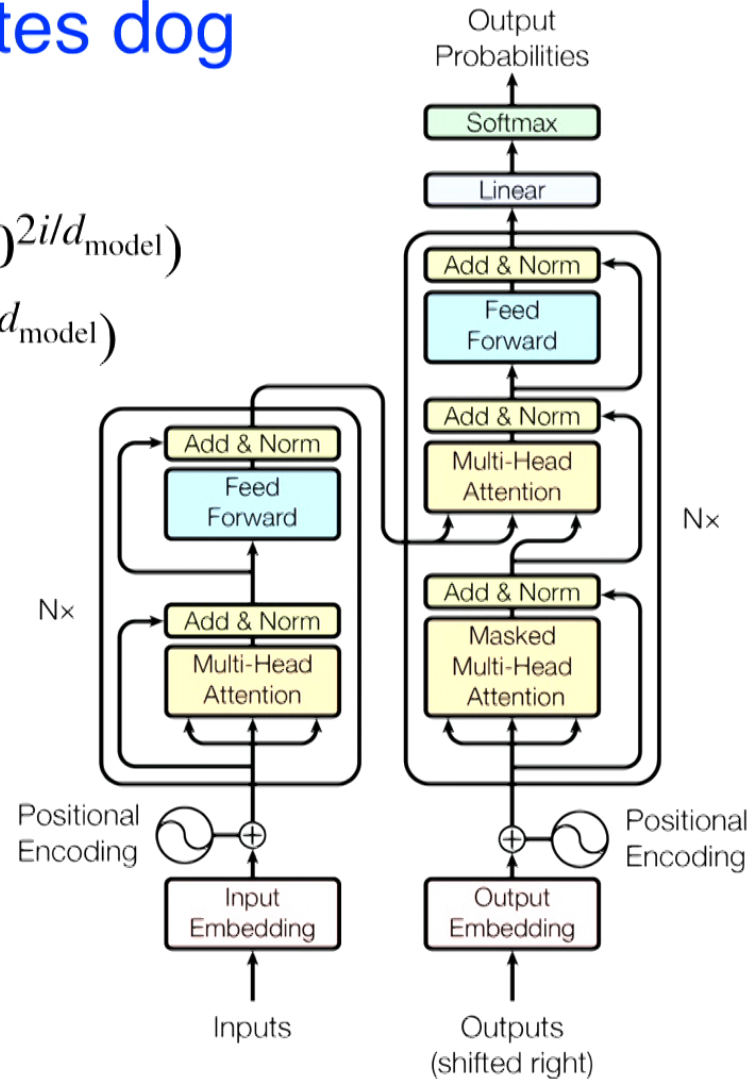
Positional Encoding:

$$PE_{(pos, 2i+1)} = \cos(pos/10000^{2i/d_{\text{model}}})$$

$$PE_{(pos, 2i)} = \sin(pos/10000^{2i/d_{\text{model}}})$$

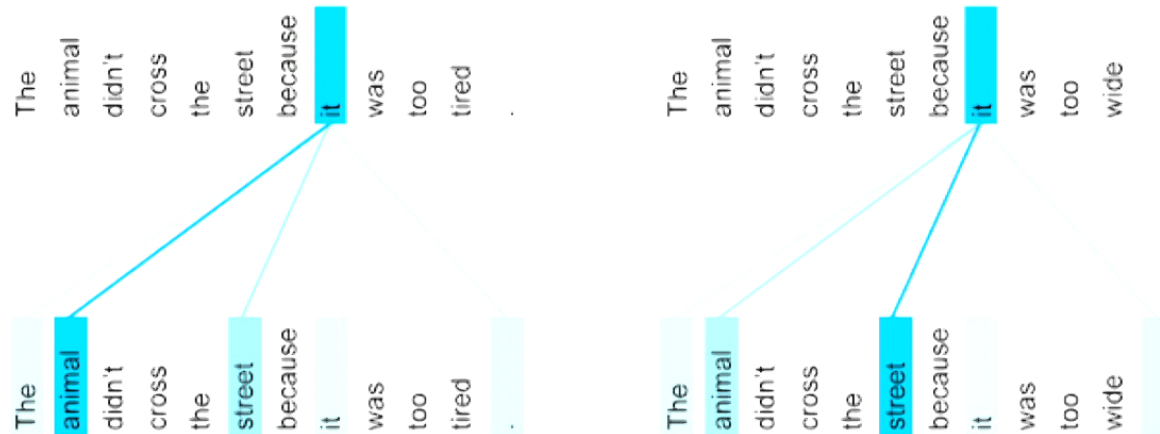
(learn to attend by relative positions)

Bytes dog Man?



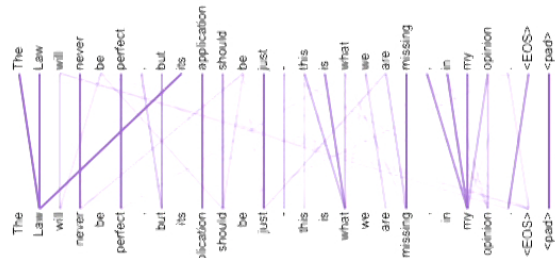
The animal didn't cross the street because **it** was too tired.  
 L'animal n'a pas traversé la rue parce qu'**il** était trop fatigué.

The animal didn't cross the street because **it** was too wide.  
 L'animal n'a pas traversé la rue parce qu'**elle** était trop large.

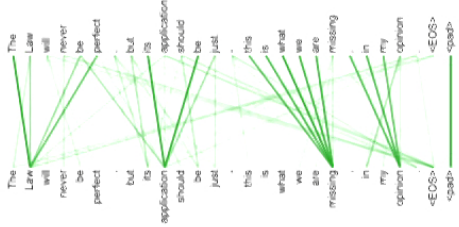
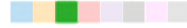


The encoder self-attention distribution for the word "it" from the 5th to the 6th layer of a Transformer trained on English to French translation (one of eight attention heads).

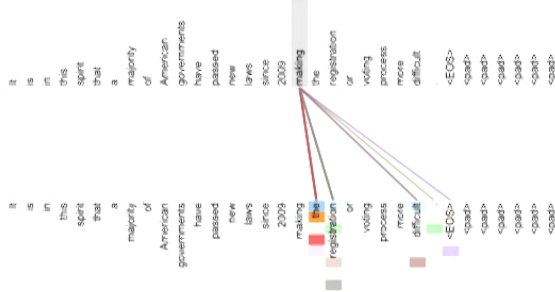
## Input-Input Layer5



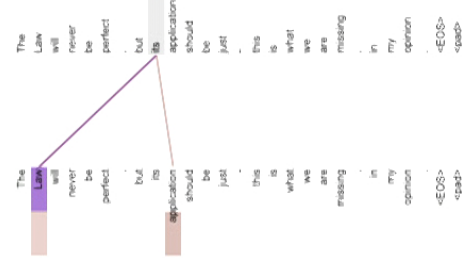
## Input-Input Layer5



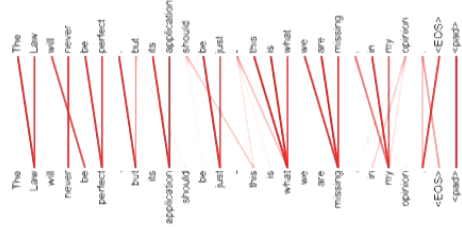
## Input-Input Layer1



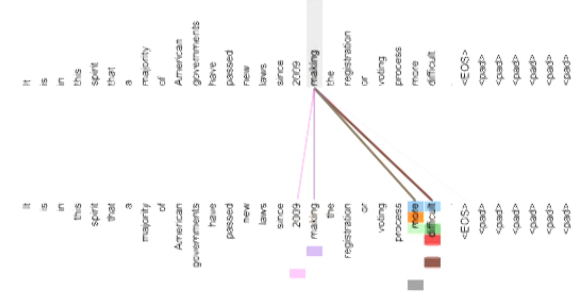
## Input-Input Layer5



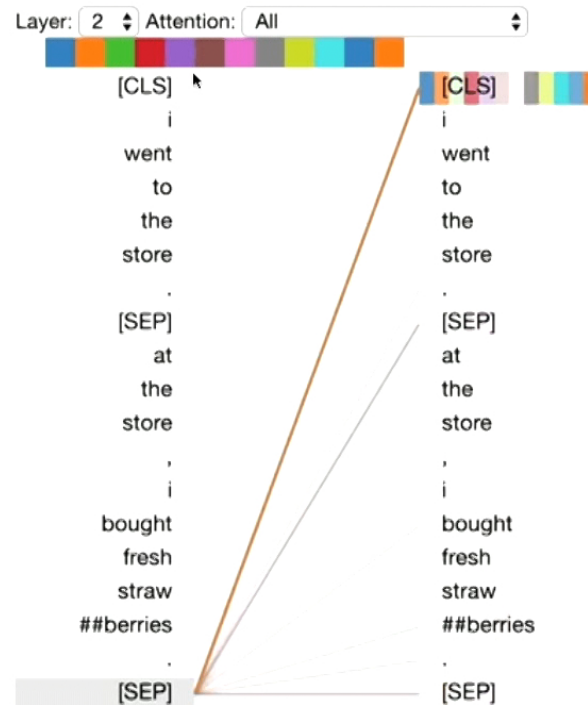
## Input-Input Layer5



## Input-Input Layer5



<https://towardsdatascience.com/deconstructing-bert-distilling-6-patterns-from-100-million-parameters-b49113672f77>



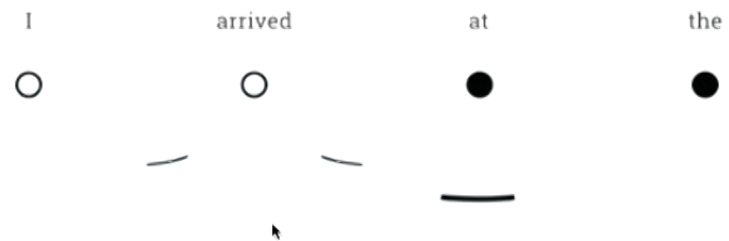
The tool visualizes attention as lines connecting the position being updated (left) with the position being attended to (right). Colors identify the corresponding attention head(s), while line thickness reflects the attention score. At the top of the tool, the user can select the model layer, as well as one or more attention heads (by clicking on the color patches at the top, representing the 12 heads).

<https://towardsdatascience.com/deconstructing-bert-distilling-6-patterns-from-100-million-parameters-b49113672f77>



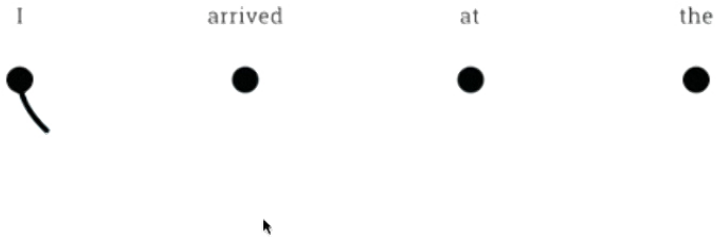
The tool visualizes attention as lines connecting the position being updated (left) with the position being attended to (right). Colors identify the corresponding attention head(s), while line thickness reflects the attention score. At the top of the tool, the user can select the model layer, as well as one or more attention heads (by clicking on the color patches at the top, representing the 12 heads).

## Encoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Encoding



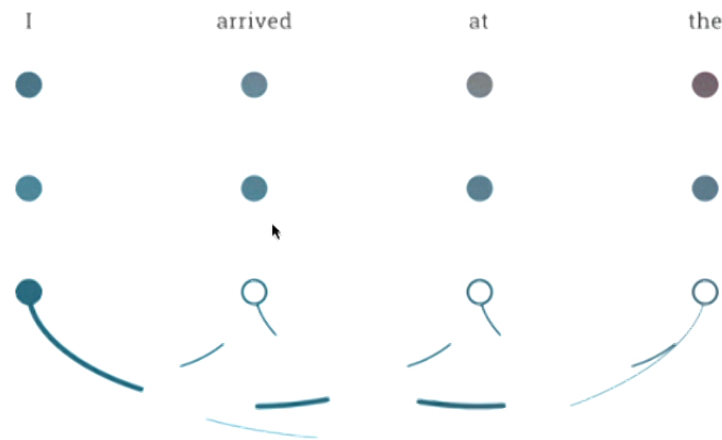
from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Encoding



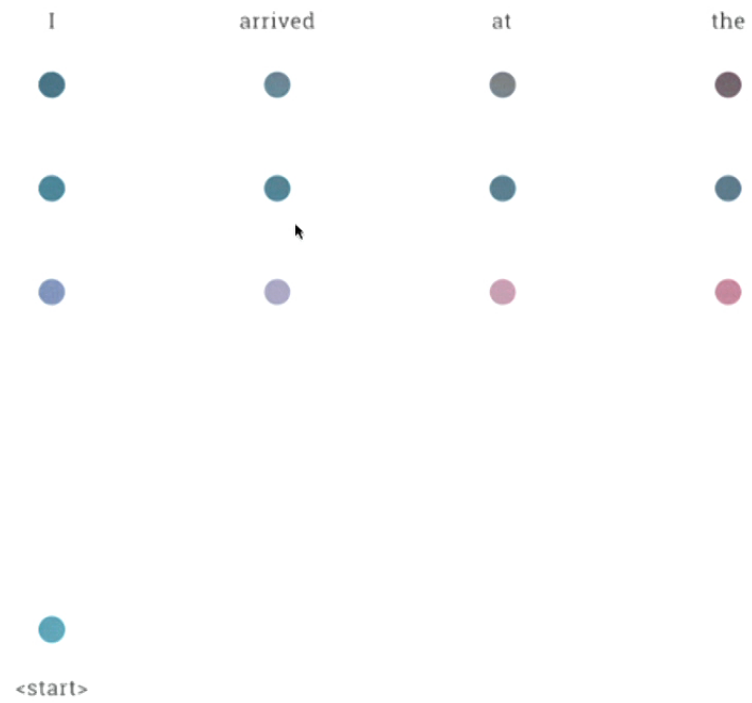
from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Encoding



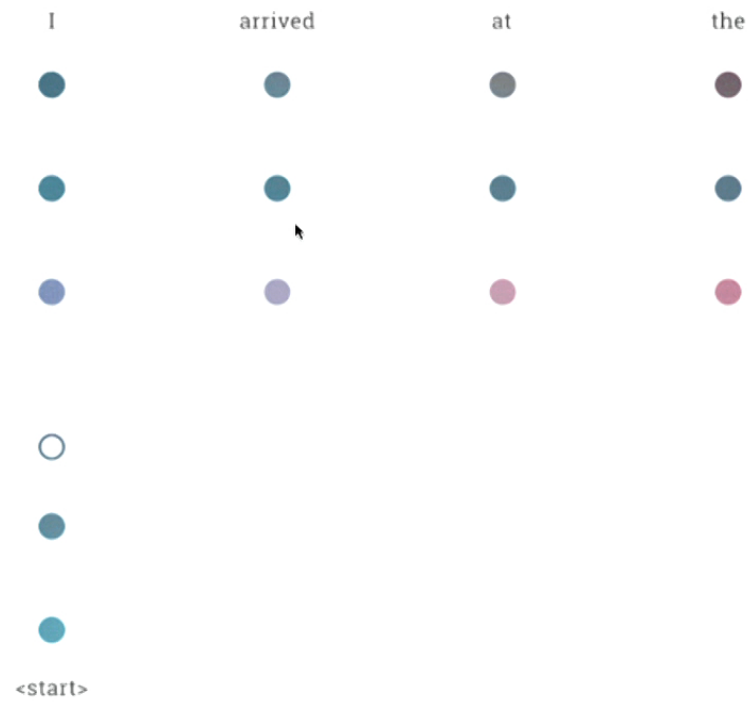
from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Decoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Decoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Decoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Decoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

## Decoding



from <https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html>

Lots of progress in 2018, generalized language models, e.g.:  
OpenAI GPT, ...

BERT = Bidirectional Encoder Representations from Transformers  
ELMo = Embeddings from Language Model

continuing in 2019, e.g.:  
GPT-2

Transformer-XL: Unleashing the Potential of Attention Models (Jan '19)  
1901.02860 Transformer-XL learns dependency that is 80% longer than RNNs  
and 450% longer than vanilla Transformers

lots of trained language models  
... (also reduce speech recog errors, reading comp, ...)

<https://openai.com/blog/better-language-models/> (Feb 2019)

## 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.

Our model, called GPT-2 (a successor to GPT = “Generative Pre-training Transformer”), was trained simply to predict the next word in 40GB of Internet text. **Due to our concerns about malicious applications of the technology**, we are not releasing the trained model. As an experiment in responsible disclosure, we are instead releasing a much smaller model for researchers to experiment with, as well as a technical paper.

GPT-2 is a large transformer-based language model with 1.5 billion parameters, trained on a dataset<sup>[1]</sup> of 8 million web pages. GPT-2 is trained with a simple objective: predict the next word, given all of the previous words within some text. The diversity of the dataset causes this simple goal to contain naturally occurring demonstrations of many tasks across diverse domains. GPT-2 is a direct scale-up of GPT, with more than 10X the parameters and trained on more than 10X the amount of data.

...

### Samples

GPT-2 generates synthetic text samples in response to the model being primed with an arbitrary input. ...

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

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

*Demonstration tutorial of retraining OpenAI's GPT-2-small (a text-generating Transformer neural network) on a large public domain Project Gutenberg poetry corpus to generate high-quality English verse.  
created: 3 March 2019; modified: 6 Jun 2019;*

OpenAI announced in February 2019 in “Better Language Models and Their Implications” their creation of “GPT-2-large”, a Transformer<sup>1</sup> neural network 10x larger than before trained (like a char-RNN with a predictive loss) by unsupervised learning on 40GB of high-quality text curated by Redditors.

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

*Demonstration tutorial of retraining OpenAI's GPT-2-small (a text-generating Transformer neural network) on a large public domain Project Gutenberg poetry corpus to generate high-quality English verse.  
created: 3 March 2019; modified: 6 Jun 2019;*

OpenAI announced in February 2019 in “Better Language Models and Their Implications” their creation of “GPT-2-large”, a Transformer<sup>1</sup> neural network 10x larger than before trained (like a char-RNN with a predictive loss) by unsupervised learning on 40GB of high-quality text curated by Redditors.

**Percy Shelley's “Ozymandias” (1818)**

I met a traveller from an antique land  
Who said: Two vast and trunkless legs of stone  
Stand in the desert... near them, on the sand,  
Half sunk, a shattered visage lies, whose frown,  
And wrinkled lip, and sneer of cold command,  
Tell that its sculptor well those passions read  
Which yet survive, stamped on these lifeless things,  
The hand that mocked them and the heart that fed;  
And on the pedestal these words appear:  
‘My name is Ozymandias, king of kings;  
Look on my works, ye Mighty, and despair!’  
Nothing beside remains. Round the decay  
Of that colossal wreck, boundless and bare  
The lone and level sands stretch far away

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

*Demonstration tutorial of retraining OpenAI's GPT-2-small (a text-generating Transformer neural network) on a large public domain Project Gutenberg poetry corpus to generate high-quality English verse.*  
created: 3 March 2019; modified: 6 Jun 2019;

OpenAI announced in February 2019 in “Better Language Models and Their Implications” their creation of “GPT-2-large”, a Transformer<sup>1</sup> neural network 10x larger than before trained (like a char-RNN with a predictive loss) by unsupervised learning on 40GB of high-quality text curated by Redditors.

#### **Percy Shelley's “Ozymandias” (1818)**

I met a traveller from an antique land  
Who said: Two vast and trunkless legs of stone  
Stand in the desert... near them, on the sand,  
Half sunk, a shattered visage lies, whose frown,  
And wrinkled lip, and sneer of cold command,  
Tell that its sculptor well those passions read  
Which yet survive, stamped on these lifeless things,  
The hand that mocked them and the heart that fed;  
And on the pedestal these words appear:  
‘My name is Ozymandias, king of kings;  
Look on my works, ye Mighty, and despair!’  
Nothing beside remains. Round the decay  
Of that colossal wreck, boundless and bare  
The lone and level sands stretch far away

The ebbing tide bears only the sands  
And trees that wave their boughs to the western breeze--  
A broken landscape, where faint breezes waft  
The lingerers on their way to Indian seas--  
A land whose shores are sunset skies,  
Where the pale mists of evening slowly sail,  
And, as we steal, the rivers murmur strange  
To the low-murmured wind at fall of night,  
And to the distant rim of the far sky:--  
Here, silent as a solitary tomb,  
With all that life can purchase and wealth can lend,  
With youth's immortal spirit, thou, O Friend!  
Art not for me to give thyself to Art?  
To thee, who art the latest breath of lands,  
Whose fame has swept the world for which men sigh,  
I give my heart up with indignant hands,  
To thee, who art the only one I seek.  
For thee, alone with life and light impearled,  
I cast my eyes from out the vast of time  
Where nothing dwells but Destiny's brief stage,  
Unshrouded, till, in its frail waste of time,  
It shall re-risen and grow once more sublime.  
And then, within the bounds where all is strife,  
And death, and life, and love, are all my own,  
I take this heart up with indignant breath,  
That seems to harden with the weight of years,  
And with the bitter hardihood of tears,  
And with the heart that has no beat or fear,  
And with the hand that has no strength to bear  
Only the world's and with the heart that fears,  
And by its sorrows only known to keep  
The beautiful and glorious years we keep.  
Here are the little walls we meant to build.  
Here stands the doorway, opening to the East,  
The child-like likeness of a poet's child

<https://towardsdatascience.com/the-fall-of-rnn-lstm-2d1594c74ce0>

# The fall of RNN / LSTM



Eugenio Culurciello in Towards Data Science [Follow](#)

Apr 13, 2018 · 8 min read

We fell for Recurrent neural networks (RNN), Long-short term memory (LSTM), and all their variants.

## Now it is time to drop them!

It is the year 2014 and LSTM and RNN make a great come-back from the dead. We all read [Colah's blog](#) and [Karpathy's ode to RNN](#). **But we were all young and unexperienced.** For a few years this was the way to solve sequence learning, sequence translation (seq2seq), which also resulted in amazing results in speech to text comprehension and the raise of [Siri](#), [Cortana](#), [Google voice assistant](#), [Alexa](#). Also let us not forget machine translation, which resulted in the ability to translate documents into different languages or [neural machine translation](#), but also translate [images into text](#), [text into images](#), and [captioning video](#), and ... well you got the idea. Then in the following years (2015–16) came [ResNet](#) and [Attention](#). One could then better understand that LSTM were a clever bypass technique. Also attention showed that MLP network could be replaced by *averaging* networks influenced by a *context vector*. More on this later.

It only took 2 more years, but today we can definitely say:

**“Drop your RNN and LSTM, they are no good!”**

But do not take our words for it, also see evidence that Attention based networks are used more and more by [Google](#), [Facebook](#), [Salesforce](#), to name a few. **All these companies have replaced RNN and variants for attention based models**, and it is just the beginning. RNN have the days counted in all applications, because they require more resources to train and run than attention-based models. See [this post](#) for more info.

<https://thegradient.pub/nlp-imagenet/>

## NLP's ImageNet moment has arrived

08.JUL.2018

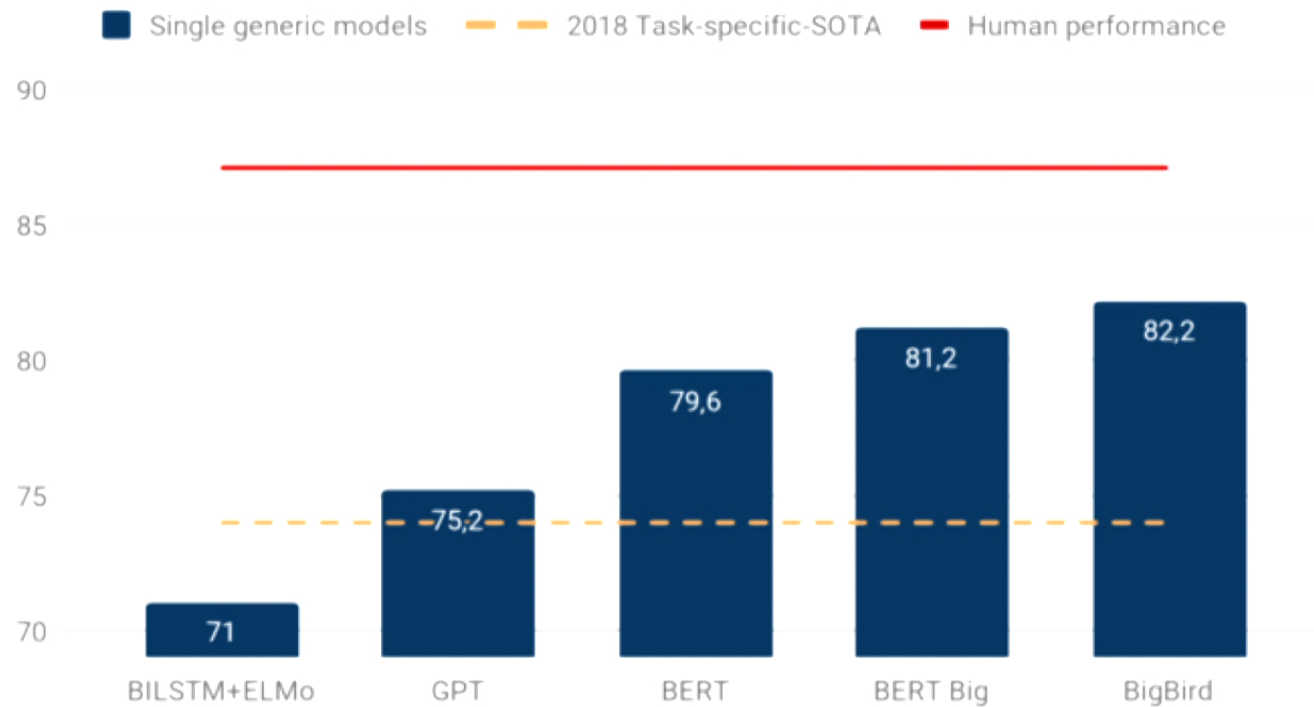
Big changes are underway in the world of Natural Language Processing (NLP).

The long reign of word vectors as NLP's core representation technique has seen an exciting new line of challengers emerge: ELMo, ULMFiT, and the OpenAI transformer. These works made headlines by demonstrating that pretrained language models can be used to achieve state-of-the-art results on a wide range of NLP tasks. Such methods herald a watershed moment: they may have the same wide-ranging impact on NLP as pretrained ImageNet models had on computer vision.

ImageNet's impact on the course of machine learning research can hardly be overstated. The dataset was originally published in 2009 and quickly evolved into the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). In 2012, the deep neural network submitted by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton performed 41% better than the next best competitor, demonstrating that deep learning was a viable strategy for machine learning and arguably triggering the explosion of deep learning in ML research.

Pretrained ImageNet models have been used to achieve state-of-the-art results in tasks such as object detection, semantic segmentation, human pose estimation, and video recognition. At the same time, they have enabled the application of CV to domains where the number of training examples is small and annotation is expensive. Transfer learning via pretraining on ImageNet is in fact so effective in CV that not using it is now considered foolhardy (Mahajan et al., 2018).

## GLUE scores evolution over 2018-2019



(GLUE = General Language Understanding Evaluation)

Chrome File Edit View History Bookmarks People Window Help

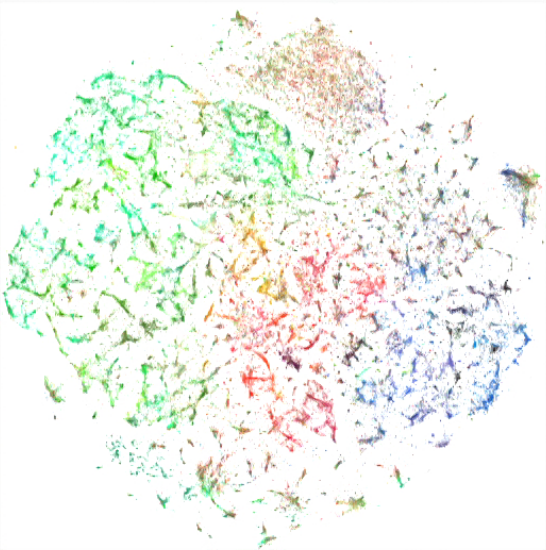
tmp/ x trigram\_gen x wvec x Deconstructing BERT: Distilling x +

Not Secure | www.cs.cornell.edu/~ginsparg/axiv/wvec.html ☆

Apps MathJax zzz zzz ac ao aq bm fc h hr meta nbm sp trn uc admin pk tur S2 twspeed lp

(Note: small-scale test based on 3 months of articles with 130k multi-gram vocab, t-sne reduced from 200 dimensional word embedding [A. Alemi and P. Ginsparg]. Zoom in to see words... WordSearch does nothing if word not in vocab.)

WordSearch:



Leaflet

