Transformer (for NLP)
Text-to-Image Creation
Vision Transformer (for CV)

Lecture by Margrit Betke, CS 585, April 16, 2024
with many slides from Steve Seitz’ videos:
Part 1 & Part 2
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red

Bob Dylan, *Tangled up in Blue*
Early one morning the sun was shining. I was laying in bed wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining. I was laying in bed wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining. I was laying in bed wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining. I was laying in bed. Her hair was still red. Wondering if she had changed at all if...
Wondering if she had changed at all, her hair still red was shining in bed. Early one morning the sun was shining, I was wondering if she had changed at all.
Wondering if she had changed at all, her hair still red, was shining. 

Early one morning the sun was shining. I lay in bed. 

Wondering if she had changed at all, her hair still red.
Early one morning the sun was shining. I was laying in bed, wondering if she had changed at all. Her hair was still red.
Early one morning the sun was shining. I wondered if she had changed at all. Her hair was still red.
Early one morning

Wondering if she had changed at all, her hair still red was shining laying in bed.

Early one morning the sun was shining I slide from Steve Seitz’s video.
Early one morning the sun was shining.

Wondering if she had changed at all,
her hair still red,
laying in bed.

Early one morning the sun was shining.
Early one morning the sun was shining. Wondering if she had changed at all, her hair still red was shining. I was laying in bed. Wondering if she had changed at all.
Early one morning the sun was shining.

Wondering if she had changed at all, her hair still red was shining in bed.

Early one morning, the sun was shining. I wonder if she had changed at all, her hair still red.

slide from Steve Seitz’s video
Early one morning the sun was shining I was 

Wondering if 

her hair still red 

laying in bed 

Early one morning the sun was shining I 

Wondering if 

she had changed at all 

slide from Steve Seitz’s video
Early one morning the sun was shining I was laying in bed

Wondering if she had changed at all her hair still red

slide from Steve Seitz's video
Early one morning the sun was shining I was laying in bed
Wondering

her hair still red

Wondering if

she had changed at all
Early one morning the sun was shining I was laying in bed
Wondering if

her hair still red
laying in bed

Early one morning the sun was shining I
Wondering if

she had changed at all
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all

slide from Steve Seitz’s video
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if

slide from Steve Seitz’s video
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red
slide from Steve Seitz’s video
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair
her hair still red

Wondering if she had changed at all
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red
Early one morning the sun was shining I was laying in bed.

Wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red

slide from Steve Seitz's video
Wondering if she had changed at all, her hair still red was shining in bed. Early one morning the sun was shining, and I slide from Steve Seitz’s video.
Wondering if she had changed at all, her hair still red, I was laying in bed. Early one morning the sun slide from Steve Seitz’s video.
the sun was shining in bed early one morning. Wondering if she had changed at all, her hair still red.
the sun was still shining

Wondering if she had changed at all

her hair still → red

Early one morning the sun was laying in bed

I slide from Steve Seitz’s video
the sun was still red

Wondering if she had changed at all
her hair still red
laying in bed
Early one morning the sun was shining

I slide from Steve Seitz’s video
Wondering if she had changed at all, her hair still red was shining. Early one morning the sun was still red in bed. laying in bed. her hair still red. Wondering if she had changed at all. slide from Steve Seitz's video.
Wondering if she had changed at all, her hair was still red. Early one morning, the sun was shining in bed. Her hair, still red, slid from Steve Seitz's video.
Wondering if she had changed at all, her hair still red was shining. Early one morning the sun was still red, her hair was laying in bed. Wondering if she had changed at all, the sun was shining.
Wondering if she had changed at all, her hair still red was shining. Early one morning the sun was still red, laying in bed, her hair still red. Wondering if she had changed at all, her hair was shining. The sun was still red.
Wondering if she had changed at all, her hair still red was shining in bed. Early one morning, the sun was laying in bed, slide from Steve Seitz’s video.
Wondering if she had changed at all her hair still red was shining laying in bed Early one morning the sun slide from Steve Seitz's video
Early one morning the sun was shining.

Wondering if she had changed at all, her hair was still red.

laying in bed
the sun was laying

Early one morning the sun was shining I

Wondering if she had changed at all

her hair still red

laying in bed
Early one morning, the sun was shining. I was laying in bed, wondering if she had changed at all. Her hair was still red.
the sun was laying in bed

Early one morning the sun was shining I

Wondering if she had changed at all

her hair still red

laying in bed
I was shining I was shining

Wondering if

her hair still red

laying in bed

Early one morning the sun was shining

Wondering if

she had changed at all
I was shining I was shining I was still red

Wondering if she had changed at all

Early one morning the sun was shining I

laying in bed

her hair still red

Wondering if she had changed at all
she was standing on the side of my mind …

side of my shoes heading out of my face …

one of my chair said our lives together …
Wondering if she had changed at all her hair still red was shining laying in bed Early one morning the sun was shining I her hair still red Wondering if she had changed at all

\[ P(x_n|x_{n-1}) \]
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red
Early one morning the sun was shining I was laying in bed. Wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining I was laying in bed wondering if she had changed at all if her hair was still red.

trigrams
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red

trigrams

slide from Steve Seitz’s video
$P(x_n | x_{n-1}, x_{n-2})$

Early one morning
one morning the morning the sun
the sun was shining
sun was shining I
shining I was laying
I was laying...

\[ P(x_n | x_{n-1}, x_{n-2}) \]
Early one morning, the sun was shining. I was laying in bed wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining I was laying in bed wondering if she had changed at all if her hair was still red
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red
\[P(x_n|x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}, x_{n-5}, x_{n-6}, x_{n-7}, x_{n-8}, x_{n-9}, x_{n-10}, x_{n-11}, x_{n-12}, x_{n-13})\]

\[10^{70} \text{ combinations}\]
Function Approximation

Fourier Series: \( f(x) = \sum + \sum + \sum + \sum + \ldots \)

Taylor Series: \( f(x) = \sum + \sum + \sum + \sum + \ldots \)

Neural Network:

\( x \rightarrow f(x) \)
$\sin(x) - \frac{x^2}{10}$
Animation shows how neural net output (red line) matches the unknown function (blue line)
$P(x_n|x_{n-1}, x_{n-2}, x_{n-3}, x_{n-4}, x_{n-5}, x_{n-6}, x_{n-7}, \ldots)$
Early one morning the sun was shining I was laying in bed
Wondering if she had changed at all if her hair was still red
Early one morning the sun was shining. I was laying in bed wondering if she had changed at all. If her hair was still red.
Early one morning the sun was shining I was laying in bed wondering if she had changed at all if her hair was still red.

eural network

slide from Steve Seitz's video
able
about
::
apex
:::
zenith
::
zygote

slide from Steve Seitz's video
word2vec
[Collobert & Weston 2008; Mikolov et al. 2013]

house, where the professor lived without his wife and child; or so he said jokingly sometimes: “Here’s where I live. My house.” His daughter often added, without resentment, for the visitor’s information, “It started out to be for me, but it’s really his.” And she might reach in to bring forth an inch-high table lamp with fluted shade, or a blue dish the size of her little fingernail, marked “Kitty” and half full of eternal milk, but she was sure to replace these, after they had been admired, pretty near exactly where they had been. The little house was very orderly, and just big enough for all it contained, though to some tastes the bric-à-brac in the parlor might seem excessive. The daughter’s preference was for the store-bought gimmicks and appliances, the toasters and carpet sweepers of Lilliput, but she knew that most adult visitors would
Word Embedding (e.g., word2Vec, GloVe)

slide from Steve Seitz’s video
Early one morning the sun was shining. I was laying in bed wondering if she had changed at all. If her hair was still red.

neural network

slide from Steve Seitz’s video
Early one morning the sun was shining I was laying in bed wondering if she had changed at all if her hair was still red.
Early one morning the sun was shining I was laying in bed wondering if she had changed at all if her hair was still red?
bed

hair was still red
Wondering if she had changed at all if her hair was still in bed.
Wondering if she had changed at all if her stillhair was bedin attention

next word prediction

red

in bed Wondering if she had changed at all if her hair was still
Wondering if she had changed at all if her hair was still bed
train
slide from Steve Seitz’s video
Two roads diverted in a yellow wood
And sorry I could not travel both
And be one traveler, long I stood
And looked down as far as I could
To where it bent in the undergrowth;

Robert Frost, *Road Not Taken*
Wondering if she had changed at all if her still hair was bedin attention

next word prediction

red

0 bed 0 0 0 0 0 0 0 0 0 0 0 0 0 0 hair was still

attention

in bed Wondering if she had changed at all if her hair was still
Wondering if she had changed at all if her still hair was bed. The next word prediction is "brown". The attention part shows the words "bed", "bed", "Wondering", "if", "she", "had", "changed", "at", "all", "if", "her", "hair", "was", "still". Slide from Steve Seitz's video.
Wondering if she had changed at all if her still hair was still

next word prediction

brown

hair was still
Wondering if she had changed at all if her still hair was red.
Wondering if she had changed at all if her hair was still in bed.

Attention slide from Steve Seitz's video.
Wondering if she had changed at all if her hair was still...
Wondering if she had changed at all if her hair was still...
Wondering if she had changed at all if her hair was still from Steve Seitz's video.
Wondering if she had changed at all if her hair was still in bed.

Attention slide from Steve Seitz's video.
Wondering if she had changed at all if her hair was still.
Wondering if she had at all if her hair was bed in attention.

slide from Steve Seitz’s video
Wondering if her hair was still the same, she wondered if she had changed at all.

\[ C_{\text{still}} = C_{\text{changed}} \]

slide from Steve Seitz’s video
Wondering if she had changed at all if her hair was still in bed.
attention

in bed Wondering if she had changed at all if her hair was still
Wondering if she had changed at all if her still hair was bed in.

slide from Steve Seitz’s video
Wondering if she had changed at all if her still hair was in bed in slide from Steve Seitz's video
It's a slide from Steve Seitz's video.
a the looking possible getting
0.4 0.3 0.1 0.1 0.1 slide from Steve Seitz’s video
It's a slide from Steve Seitz's video.
It's a lot slide from Steve Seitz's video.
It's a lot of slide from Steve Seitz's video.
It's a lot of fun to slide from Steve Seitz's video.
It's a lot of fun

prediction

attention

slide from Steve Seitz's video
The 16th president was Abraham.

Prediction:

Attention:
The 16th President was?
The capital of Zimbabwe is?
Frank Zappa’s middle name is?
Napoleon was born on this date?
The prime factorization of 19456721434 is?
Queen Victoria’s maiden name was?
US per-capita income in 1957 was?
The lat long coordinates of Rome are?
prediction

attention

96 (GPT-3) 118 (Palm)

prediction

attention

prediction

attention

slide from Steve Seitz's video
prediction

attention

prediction

attention

prediction

attention
prediction
attention

Syntax

slide from Steve Seitz's video
How much data to train?
All of it...
All text on the internet?

Is that legal?

AI & Ethics!
All text on the internet?

Is that legal?

AI & Ethics!

John Grisham, other top US authors sue OpenAI over copyrights

By Blake Brittain

September 21, 2023 6:34 AM EDT - Updated 7 months ago
All text on the internet?

Is that legal?

December 27, 2023

The Times Sues OpenAI and Microsoft Over A.I. Use of Copyrighted Work

Millions of articles from The New York Times were used to train chatbots that now compete with it, the lawsuit said.
All text on the internet?

Is that legal?

AI & Ethics!
Training the 175 billion parameters of GPT-3 on “all text on the internet” on a single GPU or computer would take 355 years and $4,600,000.
How long did it take OpenAI to train GPT-3?

a month
### What did OpenAI train on?

#### GPT-3 training data

<table>
<thead>
<tr>
<th>Dataset</th>
<th># tokens</th>
<th>Proportion within training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common Crawl</td>
<td>410 billion</td>
<td>60%</td>
</tr>
<tr>
<td>WebText2</td>
<td>19 billion</td>
<td>22%</td>
</tr>
<tr>
<td>Books1</td>
<td>12 billion</td>
<td>8%</td>
</tr>
<tr>
<td>Books2</td>
<td>55 billion</td>
<td>8%</td>
</tr>
<tr>
<td>Wikipedia</td>
<td>3 billion</td>
<td>3%</td>
</tr>
</tbody>
</table>

What about GPT-4?

• 1 trillion parameters
• Sam Altman stated that the cost of training GPT-4 was more than $100 million.

What about GPT-4?

- 1 trillion parameters
- Sam Altman stated that the cost of training GPT-4 was more than $100 million.

Why are the lawsuits so costly?

- We don’t know how to “untrain” neural networks.
- “Unlearning” is an exciting research area!

The 16th President was

The capital of Zimbabwe is

Frank Zappa’s middle name is

Napoleon was born on this date

The prime factorization of 19456721434 is

Queen Victoria’s maiden name was

US per-capita income in 1957 was

The lat long coordinates of Rome are
The 16th President was Abraham Lincoln.

The capital of Zimbabwe is Harare.

Frank Zappa’s middle name is Vincent.

Napoleon was born on this date 1769.

The prime factorization of 19456721434 is $2 \times 3 \times 3 \times 17$.

Queen Victoria’s maiden name was Alexandrina Victoria.

US per-capita income in 1957 was $2,974.

The lat long coordinates of Rome are 41.894722, 12.48.
a haiku about a turtle and a stop sign
a haiku about a turtle and a stop sign

Turtle at the stop sign
waiting for the perfect moment
to cross the busy road
translate “tangled up in blue” to 5 languages
translate “tangled up in blue” to 5 languages

1. Spanish: Enredado en azul
2. French: Emmêlé dans le bleu
3. Italian: Intrecciato nel blu
4. German: Verwickelt in blau
5. Japanese: 青い糸に絡まって
the number after the number after 37 is 74

I'm heading north. I turn right. Then I turn left. I'm now heading south
the recipe for chocolate guacamole pancakes

Ingredients:
1 ripe avocado, mashed
1/4 cup cocoa powder
1/4 cup all-purpose flour
1 teaspoon baking powder
1/4 teaspoon salt
1/2 cup milk
1 egg
1 tablespoon vegetable oil
1/4 cup chocolate chips

Directions:
1. In a medium bowl, combine the avocado, cocoa powder, flour, baking powder, and salt.
2. In a separate bowl, whisk together the milk, egg, and vegetable oil.
3. Add the wet ingredients to the dry ingredients and mix until well combined.
4. Fold in the chocolate chips.
5. Heat a large skillet over medium heat.
6. Scoop 1/4 cup batter onto the skillet for each pancake.
7. Cook for 2-3 minutes per side, or until the pancakes are golden brown.
8. Serve with your favorite toppings.
For more details: 
https://jalammar.github.io/illustrated-transformer/
https://jalammar.github.io/illustrated-gpt2/
slide from Steve Seitz’s video
a pattern of characters that looks like a star

.  o  .

o  o  o

.  o  .

slide from Steve Seitz’s video
a pattern of characters that looks like a vertical line

.  o  .
.  o  .
.  o  .
.  o  .
.  o  .
a pattern of characters that looks like a triangle

. 〇 .
. 〇 〇 .
. 〇 〇 〇 .
. 〇 〇 〇 〇 .
. 〇 〇 〇 〇 〇 .
. 〇 〇 〇 〇 〇 〇 .
slide from Steve Seitz’s video
raspberries  
pancakes  
sunsets

1 Billion

slide from Steve Seitz's video
A white image of a raspberry slide from Steve Seitz's video.
Large Language Model

slide from Steve Seitz’s video
Large Language Model
1,000,000s of pixels

1,000s of words

slide from Steve Seitz's video
slide from Steve Seitz's video
32 \times 32 = 1024
squirrel reaching for a nut

slide from Steve Seitz's video
squirrel reaching for a nut

slide from Steve Seitz's video
squirrel reaching for a nut

slide from Steve Seitz's video
squirrel reaching for a nut

slide from Steve Seitz’s video
squirrel reaching for a nut
squirrel reaching for a nut

slide from Steve Seitz's video
squirrel reaching for a nut
squirrel reaching for a nut

slide from Steve Seitz's video
squirrel reaching for a nut

slide from Steve Seitz’s video
squirrel reaching for a nut
squirrel reaching for a nut
squirrel reaching for a nut
squirrel reaching for a nut

Up-sampled 4x

slide from Steve Seitz’s video
squirrel reaching for a nut

Parti, https://parti.research.google/
squirrel reaching for a nut underwater

slide from Steve Seitz's video
fossil of a squirrel reaching for a nut
slide from Steve Seitz's video
squirrel made of toothpicks wearing sunglasses reaching for a nut

slide from Steve Seitz’s video
DLSR photograph of a whimsical fantasy house shaped like a squirrel with windows and a door, in the forest.
Squirrel reaching for a nut.  by Leonardo da Vinci

slide from Steve Seitz's video
Squirrel reaching for a nut.  Van Gogh painting

slide from Steve Seitz’s video
Intricately carved cathedral door of a squirrel reaching for a nut

slide from Steve Seitz’s video
Squirrel reaching for a nut. Woodcut tessellation pattern by M.C. Escher

slide from Steve Seitz’s video
Squirrel reaching for a nut.  Latte art

slide from Steve Seitz's video
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
Łukasz Kaiser* Google Brain lukaszkaiser@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.
Attention Is All You Need

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 2017 English-to-German translation task, improving over the existing-best results, including ensembles, by over 2 BLEU. On the WMT 2017 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.
Sequence 2 Sequence models in language

Encoder

She → is → eating → a → green → apple

Context vector (length: 5)

[0.1, -0.2, 0.8, 1.5, -0.3]

Decoder

她 → 在 → 吃 → 一个 → 绿 → 苹果

Source: https://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html
Attention and Context in language

Encoder

She → is → eating → a → green → apple

Context vector (length: 5)

[0.1, -0.2, 0.8, 1.5, -0.3]

Decoder

她 → 在 → 吃 → 一个 → 绿 → 苹果

attention
Self-Attention

- Content-based querying
- Retrieves similar items
- Weighted sum of similarities
- Constant path length between any two positions
- Variable-sized perceptive field
- Gating/multiplication enables crisp error propagation
- Trivial to parallelize (per layer)
- Can replace sequence-aligned recurrence entirely
Self-Attention Order in Machine Translation

- Encoder-Decoder Attention:
  - from output attending to words in input sequence

- Encoder Self-Attention:
  - attention to words in input sequence (all directions)

- Masked Decoder Self-Attention
  - in output attending only to words that come before
Self-Attention Order in Machine Translation

- **Encoder-Decoder Attention:**
  - from output attending to words in input sequence

- **Encoder Self-Attention:**
  - attention to words in input sequence (all directions)
  
- **Masked Decoder Self-Attention**
  - in output attending only to words that come before

You cannot use this if you are predicting the output

Use this instead!
Self-Attention Order in Machine Translation

- Encoder-Decoder Attention:
  - from output attending to words in input sequence

- Encoder Self-Attention:
  - attention to words in input sequence (all directions)

- Masked Decoder Self-Attention
  - in output attending only to words that come before

You cannot use this if you are predicting the output

Use this instead!

BUT with word-by-word processing this would take a very long time to train!
Transformer Architecture
Vaswani et al., 2017

Figure 1: The Transformer - model architecture.
Transformer Architecture

Feed forward network processes every English word

Figure 1: The Transformer - model architecture.
Transformer Architecture

Feed forward network processes every English word

Input branch

Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Positional Encoding
Input Embedding

Output branch

Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Masked Multi-Head Attention
Positional Encoding
Output Embedding

Masking: Matrix multiply: e.g. 2000 French words by 2000 French words but masking words that come afterwards with zero

English Sentence

French words, coming in

Figure 1: The Transformer - model architecture.
Transformer Architecture

Feed forward network processes every English word

Matrix multiply: e.g. processed English words by processed French words

Masking: Matrix multiply: e.g. 2000 French words by 2000 French words but masking words that come afterwards with zero

Figure 1: The Transformer - model architecture.

CS 585: Image and Video Computing
Masking Attention

Attention(Q,K,V) = softmax(Q K^T) V

Q = query vector = current English (or French) word
K key and V value = memory of words seen before
Goal: Find key(s) most similar to query and retrieve value(s) that correspond to this/these key(s)

Softmax = \sum_i e^{qk_i} / \sum_j e^{qk_j} v_i produces probability distribution over keys with peaks for keys similar to query
Masking Attention

Attention(Q,K,V) = $\text{softmax}(Q \, K^T) \, V$

Acts as a weight mask over V

Q = query vector = current English (or French) word

K key and V value = memory of words seen before

Goal: Find key(s) most similar to query and retrieve value(s) that correspond to this/these key(s)

Softmax = $\frac{\sum_i e^{q_{ki}}}{\sum_j e^{q_{kj}}} \, v_i$ produces probability distribution over keys with peaks for keys similar to query
Masking Attention

Attention(Q,K,V) = softmax(Q K^T) V

Acts as a weight mask over V

Q = query vector = current English (or French) word
K key and V value = memory of words seen before
Goal: Find key(s) most similar to query and retrieve value(s) that correspond to this/these key(s)

Softmax = \[ \frac{\sum_i e^{q_ki}}{\sum_j e^{q_kj}} \] \[ v_i \] produces probability distribution over keys with peaks for keys similar to query

Very fast:
2 matrix multiplications & 1 softmax operation
Masking Attention

Attention(Q,K,V) = softmax(Q KT/ sqrt(d_k) ) V

Q = query vector = current English (or French) word
K key and V value = memory of words seen before
Goal: Find key(s) most similar to query and retrieve value(s) that correspond to this/these key(s)

Softmax = \( \frac{\sum_i e^{q_ki}}{\sum_j e^{q_kj}} \) \( v_i \) produces probability distribution over keys with peaks for keys similar to query

Very fast:
2 matrix multiplications & 1 softmax operation

Technical detail:
sqrt(d_k) normalization needed for training
Why Multi-Head Attention?

Feed forward network processes every English word.

Matrix multiply: e.g. processed English words by processed French words.

Masking: Matrix multiply: e.g. 2000 French words by 2000 French words but masking words that come afterwards with zero.

Figure 1: The Transformer - model architecture.
Why Multi-Head Attention?

• Multiple attention layers (heads) in parallel
• Each head uses different linear transformation
• Different heads can learn different relationships
Figure 3: An example of the attention mechanism following long-distance dependencies in the encoder self-attention in layer 5 of 6. Many of the attention heads attend to a distant dependency of the verb ‘making’, completing the phrase ‘making...more difficult’. Attentions here shown only for the word ‘making’. Different colors represent different heads. Best viewed in color.
Training a Transformer

- ADAM optimizer
- Dropout during training at every layer
- Label smoothing
- Auto-regressive decoding with beam-search
- Checkpoint-averaging

- Library available: [https://github.com/tensorflow/tensor2tensor](https://github.com/tensorflow/tensor2tensor)
Transformer Architecture Complexity

• $n =$ number of words in sequence
• $d =$ network depth

Number of operations: $n^2 d$
Number of activations: $n^2 + n d$

Much better than CNNs or RNNs with number of operations $n d^2$
Transformer Architecture Complexity

• n = number of words in sequence (<70 words per sentence)
• d = network depth (maybe 1000)

Every word attends to every word

Number of operations: \( n^2 d \)  
e.g., \( 70 \times 70 \times 1000 = 4.9 \text{ mill} \)

Number of activations: \( n^2 + n \cdot d \)

Much better than CNNs or RNNs with number of operations \( n \cdot d^2 \)

e.g., \( 70 \times 1000 \times 1000 = 70 \text{ mill} \)
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

Łukasz Kaiser*
Google Brain
lukaszkaiser@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.
AN IMAGE IS WORTH 16X16 WORDS:
TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy*,†, Lucas Beyer*, Alexander Kolesnikov*, Dirk Weissenborn*,
Xiaohua Zhai*, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer,
Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby*.,†
*equal technical contribution, †equal advising
Google Research, Brain Team
{adosovitskiy, neilhoulsby}@google.com

ABSTRACT

While the Transformer architecture has become the de-facto standard for natural
language processing tasks, its applications to computer vision remain limited. In
vision, attention is either applied in conjunction with convolutional networks, or
used to replace certain components of convolutional networks while keeping their
overall structure in place. We show that this reliance on CNNs is not necessary
and a pure transformer applied directly to sequences of image patches can perform
very well on image classification tasks. When pre-trained on large amounts of
data and transferred to multiple mid-sized or small image recognition benchmarks
(ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer (ViT) attains excellent
results compared to state-of-the-art convolutional networks while requiring sub-
stantially fewer computational resources to train.
ViT Architecture

Figure 1: Model overview. We split an image into fixed-size patches, linearly embed each of them, add position embeddings, and feed the resulting sequence of vectors to a standard Transformer encoder. In order to perform classification, we use the standard approach of adding an extra learnable “classification token” to the sequence. The illustration of the Transformer encoder was inspired by Vaswani et al. (2017).
Vision Transformer (ViT)

Transformer Encoder

MLP Head

Transformer Encoder

Linear Projection of Flattened Patches

Class
Bird
Ball
Car
...

Patch + Position Embedding
* Extra learnable [class] embedding

Embedded Patches

MLP

Norm

Multi-Head Attention

Norm
Embedding Vision Transformer (ViT)

Class
Bird
Ball
Car
...

MLP Head

RGB embedding filters
(first 28 principal components)

Transformer Encoder

L x
+->

MLP

Norm

Multi-Head Attention

Norm

Embedded Patches

Linear Projection of Flattened Patches

Patch + Position Embedding
* Extra learnable [class] embedding

0 1 2 3 4 5 6 7 8 9

Embedded Patches
Position Embedding

Class
Bird
Ball
Car
...

MLP Head

Position embedding similarity

Input patch row

Input patch column

Cosine similarity

1

-1

Patch + Position Embedding

* Extra learnable [class] embedding

Linear Projection of Flattened Patches

Transformer Encoder

L x

MLP

Norm

Multi-Head Attention

Norm

Embedded Patches
Vision Transformer

<table>
<thead>
<tr>
<th>Model</th>
<th>Layers</th>
<th>Hidden size $D$</th>
<th>MLP size</th>
<th>Heads</th>
<th>Params</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT-Base</td>
<td>12</td>
<td>768</td>
<td>3072</td>
<td>12</td>
<td>86M</td>
</tr>
<tr>
<td>ViT-Large</td>
<td>24</td>
<td>1024</td>
<td>4096</td>
<td>16</td>
<td>307M</td>
</tr>
<tr>
<td>ViT-Huge</td>
<td>32</td>
<td>1280</td>
<td>5120</td>
<td>16</td>
<td>632M</td>
</tr>
</tbody>
</table>

Table 1: Details of Vision Transformer model variants.
# Vision Transformer Results

<table>
<thead>
<tr>
<th></th>
<th>Ours-JFT (ViT-H/14)</th>
<th>Ours-JFT (ViT-L/16)</th>
<th>Ours-I21k (ViT-L/16)</th>
<th>BiT-L (ResNet152x4)</th>
<th>Noisy Student (EfficientNet-L2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>88.55 ± 0.04</td>
<td>87.76 ± 0.03</td>
<td>85.30 ± 0.02</td>
<td>87.54 ± 0.02</td>
<td>88.4/88.5*</td>
</tr>
<tr>
<td>ImageNet Real</td>
<td>90.72 ± 0.05</td>
<td>90.54 ± 0.03</td>
<td>88.62 ± 0.05</td>
<td>90.54</td>
<td>90.55</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>99.50 ± 0.06</td>
<td>99.42 ± 0.03</td>
<td>99.15 ± 0.03</td>
<td>99.37 ± 0.06</td>
<td>–</td>
</tr>
<tr>
<td>CIFAR-100</td>
<td>94.55 ± 0.04</td>
<td>93.90 ± 0.05</td>
<td>93.25 ± 0.05</td>
<td>93.51 ± 0.08</td>
<td>–</td>
</tr>
<tr>
<td>Oxford-IIT Pets</td>
<td>97.56 ± 0.03</td>
<td>97.32 ± 0.11</td>
<td>94.67 ± 0.15</td>
<td>96.62 ± 0.23</td>
<td>–</td>
</tr>
<tr>
<td>Oxford Flowers-102</td>
<td>99.68 ± 0.02</td>
<td>99.74 ± 0.00</td>
<td>99.61 ± 0.02</td>
<td>99.63 ± 0.03</td>
<td>–</td>
</tr>
<tr>
<td>VTAB (19 tasks)</td>
<td>77.63 ± 0.23</td>
<td>76.28 ± 0.46</td>
<td>72.72 ± 0.21</td>
<td>76.29 ± 1.70</td>
<td>–</td>
</tr>
<tr>
<td>TPUv3-core-days</td>
<td>2.5k</td>
<td>0.68k</td>
<td>0.23k</td>
<td>9.9k</td>
<td>12.3k</td>
</tr>
</tbody>
</table>

Table 2: Comparison with state of the art on popular image classification benchmarks. We report mean and standard deviation of the accuracies, averaged over three fine-tuning runs. Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train. ViT pre-trained on the smaller public ImageNet-21k dataset performs well too. *Slightly improved 88.5% result reported in Touvron et al. (2020).
## Vision Transformer Results

<table>
<thead>
<tr>
<th>Model</th>
<th>Caltech101</th>
<th>CIFAR-100</th>
<th>DTD</th>
<th>Flowers102</th>
<th>Pets</th>
<th>Sun397</th>
<th>SVHN</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViT-H/14 (JFT)</td>
<td>95.3</td>
<td>85.5</td>
<td>75.2</td>
<td>99.7</td>
<td>97.2</td>
<td>65.0</td>
<td>88.9</td>
</tr>
<tr>
<td>ViT-L/16 (JFT)</td>
<td>95.4</td>
<td>81.9</td>
<td>74.3</td>
<td>99.7</td>
<td>96.7</td>
<td>63.5</td>
<td>87.4</td>
</tr>
<tr>
<td>ViT-L/16 (I21k)</td>
<td>90.8</td>
<td>84.1</td>
<td>74.1</td>
<td>99.3</td>
<td>92.7</td>
<td>61.0</td>
<td>80.9</td>
</tr>
</tbody>
</table>
Vision Transformer

Input

Attention

CS 585: Image and Video Computing
AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE


*equal technical contribution, †equal advising

Google Research, Brain Team
{adosovitskiy, neilhoulsby}@google.com

ABSTRACT

While the Transformer architecture has become the de-facto standard for natural language processing tasks, its applications to computer vision remain limited. In vision, attention is either applied in conjunction with convolutional networks, or used to replace certain components of convolutional networks while keeping their overall structure in place. We show that this reliance on CNNs is not necessary and a pure transformer applied directly to sequences of image patches can perform very well on image classification tasks. When pre-trained on large amounts of data and transferred to multiple mid-sized or small image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer (ViT) attains excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources to train.
Task: Image Captioning
= Creating text that describes the image

Liu et al., 2021
Animal Pose Tracking: 3D Multimodal Dataset and Token-based Pose Optimization

Patel et al., 2022
Animal Pose Tracking: 3D Multimodal Dataset and Token-based Pose Optimization

OptiPose is able to catch the subtle movement of the snout, which a simple interpolation cannot.
Animal Pose Tracking: 3D Multimodal Dataset and Token-based Pose Optimization

Patel et al., 2022

Using a Transformer:

Output:
Animal Pose Tracking: 3D Multimodal Dataset and Token-based Pose Optimization

Patel et al., 2022
Animal Pose Tracking: 3D Multimodal Dataset and Token-based Pose Optimization

Patel et al., 2022
A ConvNet for the 2020s

Zhuang Liu\textsuperscript{1,2*}, Hanzi Mao\textsuperscript{1}, Chao-Yuan Wu\textsuperscript{1}, Christoph Feichtenhofer\textsuperscript{1}, Trevor Darrell\textsuperscript{2}, Saining Xie\textsuperscript{1†}

\textsuperscript{1}Facebook AI Research (FAIR) \quad \textsuperscript{2}UC Berkeley

Code: https://github.com/facebookresearch/ConvNeXt

Abstract

The “Roaring 20s” of visual recognition began with the introduction of Vision Transformers (ViTs), which quickly superseded ConvNets as the state-of-the-art image classification model. A vanilla ViT, on the other hand, faces difficulties when applied to general computer vision tasks such as object detection and semantic segmentation. It is the hierarchical Transformers (e.g., Swin Transformers) that reintroduced several ConvNet priors, making Transformers practically viable as a generic vision backbone and demonstrating remarkable performance on a wide variety of vision tasks. However, the effectiveness of such hybrid approaches is still largely credited to the intrinsic superiority of Transformers, rather than the inherent inductive biases of convolutions. In this work, we reexamine the design spaces and test the limits of what a pure ConvNet can achieve. We gradually “modernize” a standard ResNet toward the design of a vision Transformer, and discover several key components that contribute to the performance difference along the way. The outcome of this exploration is a family of pure ConvNet models dubbed ConvNeXt. Constructed entirely from standard ConvNet modules, ConvNeXts compete favorably with Transformers in terms of accuracy and scalability, achieving 87.8% ImageNet top-1 accuracy and outperforming Swin Transformers on COCO detection and ADE20K segmentation, while maintaining the simplicity and efficiency of standard ConvNets.

![ImageNet-1K Acc.](image)

\textbf{ImageNet-1K classification results for }\textbullet\text{ConvNets and }\circ\text{vision Transformers. Each bubble’s area is proportional to FLOPs of a variant in a model family. ImageNet-1K/22K models here take 224\textsuperscript{2}/384\textsuperscript{2} images respectively. ResNet and ViT results were obtained with improved training procedures over the original papers. We demonstrate that a standard ConvNet model can achieve the same level of scalability as hierarchical vision Transformers while being much simpler in design.}

The introduction of AlexNet [40] precipitated the “ImageNet moment” [59], ushering in a new era of computer vision. The field has since evolved at a rapid speed. Representative ConvNets like VGGNet [64], Inceptions [68], ResNet(X) [28, 87], DenseNet [36], MobileNet [34], EfficientNet [71] and RegNet [54] focused on different aspects of accuracy, efficiency and scalability, and popularized many useful design principles.
A ConvNet for the 2020s

Zhuang Liu$^{1,2}$, Hanzi Mao$^1$, Chao-Yuan Wu$^1$, Christoph Feichtenhofer$^1$, Trevor Darrell$^2$, Saining Xie$^{1+}$

$^1$Facebook AI Research (FAIR)  $^2$UC Berkeley

Code: https://github.com/facebookresearch/ConNeXt

Abstract

The “Roaring 20s” of visual recognition began with the introduction of Vision Transformers (ViTs), which quickly superseded ConvNets as the state-of-the-art image classification model. A vanilla ViT, on the other hand, faces difficulties when applied to general computer vision tasks such as object detection and semantic segmentation. It is the hierarchical Transformers (e.g., Swin Transformers) that re-introduced several ConvNet priors, making Transformers practically viable as a general vision backbone and demonstrating remarkable performance on a wide variety of vision tasks. However, the effectiveness of such hybrid approaches is still largely credited to the intrinsic superiority of Transformers, rather than the inherent inductive biases of convolutions. In this work, we re-examine the design spaces and test the limits of what a pure ConvNet can achieve. We gradually “modernize” a standard ResNet toward the design of a vision Transformer, and discover several key components that contribute to the performance difference along the way. The outcome of this exploration is a family of pure ConvNet models dubbed ConNeXt. Constructed entirely from standard ConvNet modules, ConNeXts compete favorably with Transformers in terms of accuracy and scalability, achieving 87.8% ImageNet top-1 accuracy and outperforming Swin Transformers on COCO detection and ADE20K segmentation, while maintaining the simplicity and efficiency of standard ConvNets.

Figure 1. ImageNet-1K classification results for • ConNeXt and • vision Transformers. Each bubble’s area is proportional to FLOPs of a variant in a model family. ImageNet-1K/22K models here take 224$^2$/384$^2$ images respectively. ResNet and ViT results were obtained with improved training procedures over the original papers. We demonstrate that a standard ConvNet model can achieve the same level of scalability as hierarchical vision Transformers while being much simpler in design.

The introduction of AlexNet [40] precipitated the “ImageNet moment” [59], ushering in a new era of computer vision. The field has since evolved at a rapid speed. Representative ConvNets like VGGNet [64], Inceptions [68], ResNet(X) [28, 87], DenseNet [36], MobileNet [34], EfficientNet [71] and RegNet [54] focused on different aspects of accuracy, efficiency and scalability, and popularized many useful design principles.
Learning Outcomes

Understand

• Concept of attention
• Transformers for NLP ("Attention is All you Need")
• Vision transformers for object recognition ("An image is worth 16x16 Words")
• Vision transformers for image captioning
• Vision transformers for 3D pose optimization and tracking