CS 188: Artificial Intelligence

Advanced Topics: Natural Language Processing

Instructors: Eve Fleisig & Evgeny Pobachienko

[Slides courtesy of Dan Klein, Abigail See, Greg Durrett, Yejin Choi, John DeNero, Eric Wallace, Kevin Lin, Fei-Fei Li, Sergey Levine, Pieter Abbeel, Jay Alammar, and many others]
Natural Language Processing

Goal: Deep Understanding
- Requires knowledge, context, and grounding
- Just starting to see successes

Reality: Shallow Matching
- Requires robustness and scale
- Amazing successes, but fundamental limitations

NLP
grep
What tasks do we care about?

Applications
- Text generation
- Chatbots
- Machine translation
- Web search
- Speech recognition
- Virtual assistants
- Text-to-speech
- Syntactic parsing, semantic parsing, semantic role labeling, word sense induction, coreference resolution, summarization, sentiment analysis...
Disney décide de changer le nom du légendaire studio de cinéma Fox

Le nouveau propriétaire va faire disparaître la mention « Fox » à la 20th Century pour que le nom du studio ne soit pas associé à l’image ultra-conservatrice de la chaîne de télévision Fox News.

Révolution dans le monde d'Hollywood. Le célèbre studio de cinéma 20th Century Fox va voir une partie de son nom modifiée et devenir 20th Century Studios, décision prise par son nouveau propriétaire, Disney.

Selon plusieurs médias américains, Disney aurait décidé de modifier le nom du studio pour qu'il ne soit plus associé à la grande chaîne Fox mais aussi et surtout à Fox News, la chaîne d'information en continu.
How many US states capitals' are also their largest cities?

As of my last update in September 2021, there were eight U.S. states where the capital city is also the largest city in the state. These states are:

1. Atlanta, Georgia
2. Boston, Massachusetts
3. Denver, Colorado
4. Honolulu, Hawaii
5. Indianapolis, Indiana
6. Oklahoma City, Oklahoma
7. Phoenix, Arizona
8. Salt Lake City, Utah

Please note that this information might have changed since then due to population shifts and other factors. For the most up-to-date information, it's best to consult a recent source or check official government websites.
Jeopardy!
Question Answering: Watson

"a camel is a horse designed by"

Wikipedia

a camel is a horse designed by a committee

The Phrase Finder

A camel is a horse designed by committee

Posted by Ruben P. Mendez on April 16, 2004

Does anyone know the origin of this maxim? I heard it way back at the United Nations, which is chockfull of committees. It may have originated there, but I'd like an authoritative explanation. Thanks
US Cities: Its largest airport is named for a World War II hero; its second largest, for a World War II battle.
Language Comprehension?

Opera refers to a dramatic art form, originating in Europe, in which the emotional content is conveyed to the audience as much through music, both vocal and instrumental, as it is through the lyrics. By contrast, in musical theater an actor’s dramatic performance is primary, and the music plays a lesser role. The drama in opera is presented using the primary elements of theater such as scenery, costumes, and acting. However, the words of the opera, or libretto, are sung rather than spoken. The singers are accompanied by a musical ensemble ranging from a small instrumental ensemble to a full symphonic orchestra.

1. It is pointed out in the reading that opera ----.

- A) has developed under the influence of musical theater
- B) is a drama sung with the accompaniment of an orchestra
- C) is not a high-budget production
- D) is often performed in Europe
- E) is the most complex of all the performing arts
Conversations with Devices?

"What's the best movie to see this weekend?"
That would probably start an argument. But here's a list of highly-regarded movies:

- NORTH BY NORTHWEST
  Released July 17, 1959
  100%

- THE TREASURE OF THE SIERRA
  Released January 6, 1948
  100%

"What can I help you with?"
"You need to start understanding me Siri."
I'll make a note of that.
"Yeah, you better make a note of that."

Noted:
Of that
ELIZA: You're not really talking about me -- are you?
YOU: What?
ELIZA: What do you think?
YOU: I mean, I feel happy to be writing right now, to be typing this as we speak. Lately, I've had writer's block.
ELIZA: You say you mean?
YOU: That's what I said.
ELIZA: Can you elaborate on that?
YOU: I only really feel happy when I'm writing.
ELIZA: How long have you been writing?
YOU: Always, but I've been writing seriously since I was eighteen.
ELIZA: What incident are you thinking of?
YOU: Um.
ELIZA: What does that suggest to you?
YOU: I don't think it's enough, to only be happy when you write.
ELIZA: Does that trouble you?
YOU: ...Um, because you can't write all the time. And then there's the real world, and you still have to deal with real people.
ELIZA: What are your feelings now?
YOU: I've been thinking a lot of that poem by Yeats: "The Man and the Echo." Do you know it?
ELIZA: Do you say you've been thinking a lot of that poem by yeats: "the man and the echo for some special reason?
YOU: Yes.
Large Language Models

ChatGPT PLUS

Give me ideas for what to do with my kids' art
Create a workout plan for resistance training
Show me a code snippet of a website's sticky header
Recommend a dish to bring to a potluck
Send a message

ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT_A Jul 20 Version
Approach #1: Lexical Translation

**Step #1: Learn Alignments**
- Learn mappings between words in source and target language
- IBM Model 1, 2, 3, 4, 5...
- Can also learn a phrase table of mappings

**Step #2: Generate Language**
- Search problem over the space of natural language strings
- Can use approaches like A* to guide search
Stevie Wonder announces he’ll be having kidney surgery during London concert

By Amir Vera, CNN
Updated 11:16 PM EDT, Sat July 06, 2019

(CNN) — Stevie Wonder will be taking a break from music.

The legendary singer-songwriter announced during a concert in London Saturday that he will be undergoing kidney surgery.
Issue: Ambiguities

- **Headlines:**
  - Enraged Cow Injures Farmer with Ax
  - Teacher Strikes Idle Kids
  - Hospitals Are Sued by 7 Foot Doctors
  - Ban on Nude Dancing on Governor’s Desk
  - Iraqi Head Seeks Arms
  - Stolen Painting Found by Tree
  - Kids Make Nutritious Snacks
  - Local HS Dropouts Cut in Half

- Can we come up with a representation to disambiguate the two readings of each headline?
We Need Representation: Linguistic Structure

- **Teacher Strikes Idle Kids**
  - N N V N
  - N V ADJ N

- **Iraqi Head Seeks Arms**
  - body/ position
  - body/ weapon

- **Ban on Nude Dancing on Governor’s Desk**
  - NP
    - PP
      - NP
        - N P
          - PP
            - N
              - PP
                - NP

- Syntactic and semantic ambiguities: parsing needed to resolve these, but need context to figure out which parse is correct
Approach #2: Predict Intermediate Structures

He adores listening to music

Syntax-based Statistical Machine Translation

Kare ha ongaku wo kiku no ga daisuki desu

Image courtesy of https://vas3k.com/blog/machine_translation/
Approach #3: Language Modeling

- the station signs are in deep in english
- the stations signs are in deep in english
- the station signs are in deep into english
- the station 's signs are in deep in english
- the station signs are in deep in the english
- the station signs are indeed in english
- the station 's signs are indeed in english
- the station signs are indians in english
Noisy Channel Model: ASR

- We want to predict a sentence given acoustics:

\[ w^* = \arg \max_w P(w|a) \]

- The noisy-channel approach:

\[ w^* = \arg \max_w P(w|a) \]

\[ = \arg \max_w \frac{P(a|w)P(w)}{P(a)} \]

\[ \propto \arg \max_w P(a|w)P(w) \]

Acoustic model: score fit between sounds and words

Language model: score plausibility of word sequences
“Having guessed and inferred considerably about, the powerful new mechanized methods in cryptography...one naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography. When I look at an article in Russian, I say: ‘This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.’ ”

Warren Weaver (1947)
Disney decides to change the name of the legendary Fox film studio

The new owner will remove the mention “Fox” in the 20th Century so that the name of the studio is not associated with the ultra-conservative image of the television channel Fox News.

Revolution in the Hollywood world. The famous film studio 20th Century Fox will see part of its name changed and become 20th Century Studios, decision taken by its new owner, Disney.

According to several American media, Disney has decided to change the name of the studio so that it is no longer associated with the big chain Fox but also and especially with Fox News, the news channel.
Empirical N-Grams

- Use statistics from data (examples here from Google N-Grams)

\[
\hat{P}(\text{door}|\text{the}) = \frac{14112454}{23135851162} = 0.0006
\]

- This is the maximum likelihood estimate, which needs modification
- N-gram models use such counts to compute probabilities on demand
Smoothing

- We often want to make estimates from sparse statistics:

\[
P(w \mid \text{denied the}) \quad \begin{array}{l}
3 \text{ allegations} \\
2 \text{ reports} \\
1 \text{ claims} \\
1 \text{ request} \\
7 \text{ total}
\end{array}
\]

- Smoothing flattens spiky distributions so they generalize better:

\[
P(w \mid \text{denied the}) \quad \begin{array}{l}
2.5 \text{ allegations} \\
1.5 \text{ reports} \\
0.5 \text{ claims} \\
0.5 \text{ request} \\
2 \text{ other} \\
7 \text{ total}
\end{array}
\]

- Very important all over NLP, but easy to do badly
Please close the first door on the left.

<table>
<thead>
<tr>
<th>4-Gram</th>
<th>3-Gram</th>
<th>2-Gram</th>
</tr>
</thead>
<tbody>
<tr>
<td>3380 please close the door</td>
<td>197302 close the window</td>
<td>198015222 the first</td>
</tr>
<tr>
<td>1601 please close the window</td>
<td>191125 close the door</td>
<td>194623024 the same</td>
</tr>
<tr>
<td>1164 please close the new</td>
<td>152500 close the gap</td>
<td>168504105 the following</td>
</tr>
<tr>
<td>1159 please close the gate</td>
<td>116451 close the thread</td>
<td>158562063 the world</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0 please close the first</td>
<td>8662 close the first</td>
<td>...</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>13951 please close the *</td>
<td>3785230 close the *</td>
<td>23135851162 the *</td>
</tr>
</tbody>
</table>

0.0 0.002 0.009

Specific but Sparse ⇔ Dense but General

$$\lambda \hat{P}(w|w_{-1}, w_{-2}) + \lambda' \hat{P}(w|w_{-1}) + \lambda'' \hat{P}(w)$$
Discounting

- Observation: N-grams occur more in training data than they will later

Empirical Bigram Counts (Church and Gale, 91)

<table>
<thead>
<tr>
<th>Count in 22M Words</th>
<th>Future c* (Next 22M)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

- Absolute discounting: reduce counts by a small constant, redistribute “shaved” mass to a model of new events

\[
P_{ad}(w|w') = \frac{c(w', w) - d}{c(w')} + \alpha(w') \hat{P}(w)
\]
Reminder: Feedforward Neural Nets

\[ P(y|x) = \text{softmax}(Wg(Vf(x))) \]

- **f(x)**: Input function
- **V**: \(d \times n\) matrix
- **g**: Nonlinearity (tanh, relu, ...)
- **W**: \(num\_classes \times d\) matrix
- **P(y|x)**: Output probabilities

**Diagram Notes**:
- \(d\) hidden units
- \(n\) features
- \(num\_classes\) classes
A Feedforward N-Gram Model?
Early Neural Language Models

- Fixed-order feed-forward neural LMs
  - Eg Bengio et al 03
  - Allow generalization across contexts in more nuanced ways than prefixing
  - Allow different kinds of pooling in different contexts
  - Much more expensive to train
Recurrent NNs
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk.
Today, I went to the store and bought some milk and
Today, I went to the store and bought some milk and eggs.
Today, I went to the store and bought some milk and eggs. I
Today, I went to the store and bought some milk and eggs. I knew
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was
Today, I went to the store and bought some milk and eggs. I knew it was going
Today, I went to the store and bought some milk and eggs. I knew it was going to
Today, I went to the store and bought some milk and eggs. I knew it was going to rain,
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella,
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet.
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the
Today, I went to the store and bought some milk and eggs. I knew it was going to rain, but I forgot to take my umbrella, and ended up getting wet on the way.
Language Modeling

Input
Features

Trained Language Model

Task:
Predict the next word

Output
Prediction

0% aardvark
0% aarhus
0.1% aaron
...
40% not
...
0.01 zyzzyva

Source: Jay Alammar
Recall: Language Modeling

- Goal: learn a probability distribution over possible next words
  \[ P(w_k \mid w_{k-1}, \ldots, w_0) \]

- Markovian assumption (used in n-gram models):
  \[ P(w_k \mid w_{k-1}, \ldots, w_0) = P(w_k \mid w_{k-1}, \ldots, w_{k-n+1}) \]
  
  - E.g., in a bigram model: \[ P(w_k \mid w_{k-1}, \ldots, w_0) = P(w_k \mid w_{k-1}, w_{k-2}) \]
RNNs

- Feedforward NNs can’t handle variable length input: each position in the feature vector has fixed semantics

```
the   movie   was   great
```

```
that  was    great   !
```

- These don’t look related (*great* is in two different orthogonal subspaces)

- Instead, we need to:
  1) Process each word in a uniform way
  2) ...while still exploiting the context that that token occurs in
- Cell that takes some input $x$, has some hidden state $h$, and updates that hidden state and produces output $y$ (all vector-valued)
RNN Uses

- **Transducer**: make some prediction for each element in a sequence
  
  output $y = \text{score for each tag, then softmax}$

  ![Diagram](image)

  the movie was great

- **Acceptor/encoder**: encode a sequence into a fixed-sized vector and use that for some purpose
  
  ![Diagram](image)

  the movie was great

  predict sentiment (matmul + softmax)

  translate

  paraphrase/compress
Basic RNNs

\[ h_t = \tanh(W x_t + V h_{t-1} + b_h) \]

- Updates hidden state based on input and current hidden state

\[ y_t = \tanh(U h_t + b_y) \]

- Computes output from hidden state

- Long history! (invented in the late 1980s)

Elman (1990)
Training RNNs

- "Backpropagation through time": build the network as one big computation graph, some parameters are shared.
- RNN potentially needs to learn how to "remember" information for a long time!

The movie was great.

It was my favorite movie of 2016, though it wasn’t without problems. 

"Correct" parameter update is to do a better job of remembering the sentiment of favorite.
Problem: Vanishing Gradients

- Contribution of earlier inputs decreases if matrices are contractive (first eigenvalue < 1), non-linearities are squashing, etc.
- Gradients can be viewed as a measure of the effect of the past on the future.
- That’s a problem for optimization but also means that information naturally decays quickly, so model will tend to capture local information.
Core Issue: Information Decay

- The main problem is that *it’s too difficult for the RNN to learn to preserve information over many timesteps.*

- In a vanilla RNN, the hidden state is constantly being rewritten

\[ h^{(t)} = \sigma \left( W_h h^{(t-1)} + W_x x^{(t)} + b \right) \]

- How about a RNN with separate memory?
Problem: Exploding Gradients

- Gradients can also be too large
  - Leads to overshooting / jumping around the parameter space
  - Common solution: gradient clipping
Key Idea: Propagated State

- Information decays in RNNs because it gets multiplied each time step.
- Idea: have a channel called the cell state that by default just gets propagated (the “conveyor belt”)
- Gates make explicit decisions about what to add / forget from this channel.
LSTMs
LSTMs

Forget some cell content

Write some new cell content

Output some cell content to the hidden state

Compute the forget gate

Compute the input gate

Compute the new cell content

Compute the output gate
LSTMs

- Ignoring recurrent state entirely:
  - Lets us get feedforward layer over token
- Ignoring input:
  - Lets us discard stopwords
- Summing inputs:
  - Lets us compute a bag-of-words representation
Gradient still diminishes, but in a controlled way and generally by less — usually initialize forget gate = 1 to remember everything to start.

http://colah.github.io/posts/2015-08-Understanding-LSTMs/
The Bottleneck Problem

Encoding of the source sentence.

Target sentence (output)

Source sentence (input)

il a m' entarté

<START> he hit me with a pie <END>

he hit me with a pie
LSTMs with Attention

On this decoder timestep, we’re mostly focusing on the first encoder hidden state (“he”)

Take softmax to turn the scores into a probability distribution

Source sentence (input)
LSTMs with Attention

Use the attention distribution to take a weighted sum of the encoder hidden states.

The attention output mostly contains information from the hidden states that received high attention.
LSTMs with Attention

Concatenate attention output with decoder hidden state, then use to compute $\hat{y}_1$ as before.
A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.
Self-Attention
Self-Attention

Source: Jay Alammar
Self-Attention

Source: Jay Alammar
1) For each input token, create a **query vector**, a **key vector**, and a **value vector** by multiplying by weight Matrices $W^q$, $W^k$, $W^v$.
2) Multiply (dot product) the current query vector, by all the key vectors, to get a score of how well they match.

Self-Attention
3) Multiply the value vectors by the scores, then sum up
Self-Attention

\[
\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) = Z
\]

Source: Jay Alammar
Multi-Head Attention

Source: Jay Alammar
Multi-Head Attention

1) Concatenate all the attention heads

\[ Z_0 \quad Z_1 \quad Z_2 \quad Z_3 \quad Z_4 \quad Z_5 \quad Z_6 \quad Z_7 \]

2) Multiply with a weight matrix \( W_0 \) that was trained jointly with the model

\[ X \]

3) The result would be the \( Z \) matrix that captures information from all the attention heads. We can send this forward to the FFNN

\[ Z \]

Source: Jay Alammar
Transformers

Instead of an RNN, just use attention

High throughput & expressivity: compute queries, keys and values as (different) linear transformations of the input.

Attention weights are queries • keys; outputs are sums of weighted values.

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

Source: Jay Alammar
Transformer

INPUT
Je suis étudiant

THE TRANSFORMER

OUTPUT
I am a student

Source: Jay Alammar
Transformer

Source: Jay Alammar
Transformer Input

**EMBEDDING WITH TIME SIGNAL**

\[ \mathbf{x}_1 \quad = \quad \mathbf{t}_1 \quad + \quad \mathbf{x}_2 \quad = \quad \mathbf{t}_2 \quad + \quad \mathbf{x}_2 \]

**POSITIONAL_encoding**

**EMBEDDINGS**

**INPUT**

\[ \text{Je} \quad \text{suis} \]

Source: Jay Alammar
Transformer Encoder

Source: Jay Alammar
Full Transformer: Adding the Decoder

Source: Jay Alammar
Masked Language Models

Key idea: learn representations and then fine-tune (training ≠ inference)
Use the output of the masked word’s position to predict the masked word.

Randomly mask 15% of tokens.

Input

Source: Jay Alammar
Autoregressive Language Models

Key idea: learn next-word prediction directly (training = inference)
Masked vs. Autoregressive Language Modeling

\[
\log p(\bar{\mathbf{x}}|\hat{\mathbf{x}}) = \sum_{t=1}^{T} \text{mask}_t \log p(x_t|\hat{\mathbf{x}})
\]

\[
\log p(\mathbf{x}) = \sum_{t=1}^{T} \log p(x_t|\mathbf{x}_{<t})
\]
Pretraining & Fine-tuning

Pre-training

- President Franklin born January 1882.
- Lily couldn't. The waitress had brought the largest of chocolate cake seen.
- Our hand-picked and sun-dried orchard in Georgia.
- D. Roosevelt was in believe her eyes piece she had ever
- peaches are at our
Pretraining & Fine-tuning

T5

President Franklin &<M> born &<M> January 1882.

Lily couldn't &<M>. The waitress had brought the largest &<M> of chocolate cake &<M> seen.

Our &<M> hand-picked and sun-dried &<M> orchard in Georgia.

D. Roosevelt was &<M> in believe her eyes &<M> piece &<M> she had ever

peaches are &<M> at our

Pre-training

Fine-tuning

When was Franklin D. Roosevelt born?

1882
Unsupervised objective

Supervised objective
"translate English to German: That is good."

cola sentence: The course is jumping well.

"stsb sentence1: The rhino grazed on the grass. sentence2: A rhino is grazing in a field."

"summarize: state authorities dispatched emergency crews tuesday to survey the damage after an onslaught of severe weather in mississippi..."

"Das ist gut."

"not acceptable"

"3.8"

"six people hospitalized after a storm in attala county."
Can you translate "language models are powerful these days" into French?

Certainly! The translation of "language models are powerful these days" into French is:

"Les modèles de langage sont puissants de nos jours."

Can you write a syntactic parse for the sentence "Kids Make Nutritious Snacks"?

Sure! Here's a syntactic parse for the sentence "Kids Make Nutritious Snacks":

```
(S
  (NP (NNS Kids))
  (VP (VBZ Make))
    (NP (JJ Nutritious) (NNS Snacks)))
```
Few- & Zero-Shot Learning

**Few-shot**

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

1. Translate English to French:
   - sea otter => loutre de mer
   - peppermint => menthe poivrée
   - plush giraffe => girafe peluche
   - cheese => 

   **task description**
   **examples**
   **prompt**
Few- & Zero-Shot Learning

Few-shot
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

Translate English to French:
- sea otter => loutre de mer
- peppermint => menthe poivrée
- plush giraffe => girafe peluche
- cheese =>

One-shot
In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

1. Translate English to French:
2. sea otter => loutre de mer
3. cheese =>
Few- & Zero-Shot Learning

**Few-shot**
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

**Zero-shot**
The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

**One-shot**
In addition to the task description, the model sees a single example of the task. No gradient updates are performed.
Instruction Tuning and RLHF

Key issue: language modeling ≠ assisting users

<table>
<thead>
<tr>
<th>PROMPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explain the moon landing to a 6 year old in a few sentences.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COMPLETION</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-3</td>
</tr>
<tr>
<td>Explain the theory of gravity to a 6 year old.</td>
</tr>
<tr>
<td>Explain the theory of relativity to a 6 year old in a few sentences.</td>
</tr>
<tr>
<td>Explain the big bang theory to a 6 year old.</td>
</tr>
<tr>
<td>Explain evolution to a 6 year old.</td>
</tr>
</tbody>
</table>
Instruction Tuning and RLHF

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Explain the moon landing to a 6 year old in a few sentences.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completion</td>
<td>GPT-3</td>
</tr>
<tr>
<td></td>
<td>Explain the theory of gravity to a 6 year old.</td>
</tr>
<tr>
<td></td>
<td>Explain the theory of relativity to a 6 year old in a few sentences.</td>
</tr>
<tr>
<td></td>
<td>Explain the big bang theory to a 6 year old.</td>
</tr>
<tr>
<td></td>
<td>Explain evolution to a 6 year old.</td>
</tr>
<tr>
<td>InstructGPT</td>
<td>People went to the moon, and they took pictures of what they saw, and sent them back to the earth so we could all see them.</td>
</tr>
</tbody>
</table>
Reinforcement Learning from Human Feedback

Step 1

**Collect demonstration data, and train a supervised policy.**

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.
Reinforcement Learning from Human Feedback

Step 1: Collect demonstration data, and train a supervised policy.
- A prompt is sampled from our prompt dataset.
- A labeler demonstrates the desired output behavior.
- This data is used to fine-tune GPT-3 with supervised learning.

Step 2: Collect comparison data, and train a reward model.
- A prompt and several model outputs are sampled.
- A labeler ranks the outputs from best to worst.
- This data is used to train our reward model.
Reinforcement Learning from Human Feedback

Step 1
Collect demonstration data, and train a supervised policy.

A prompt is sampled from our prompt dataset.

A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

Step 2
Collect comparison data, and train a reward model.

A prompt and several model outputs are sampled.

A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3
Optimize a policy against the reward model using reinforcement learning.

A new prompt is sampled from the dataset.

The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.
Scaling Data & Compute

Kaplan et al., 2020; Hoffmann et al., 2022
Going Forward

- Tool use (e.g., getting language models to use APIs)
- Grounding into non-linguistic inputs (e.g., vision, sensor data, etc.)
- Managing security & privacy
- Efficient / on-device / smaller / faster models
- Avoiding harmful or undesirable outputs
- Supporting multilinguality, esp. for low resource languages
Bonus: Computer Vision
What tasks do we care about?

- Object detection and classification
- Semantic segmentation
- Image captioning
- Visual question answering
- Video classification and understanding
- Image generation
- ...
Image Classification

cat
dog
horse
person
airplane
house
...

![Cat Image]
Beyond Image Classification

Classification

No spatial extent

Semantic Segmentation

No objects, just pixels

Object Detection

Multiple Object

Instance Segmentation

This image is CC0 public domain
**Image Generation**

<table>
<thead>
<tr>
<th>TEXT PROMPT</th>
<th>an armchair in the shape of an avocado, an armchair imitating an avocado.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AI-GENERATED IMAGES</td>
<td><img src="image1.png" alt="Image 1" /> <img src="image2.png" alt="Image 2" /> <img src="image3.png" alt="Image 3" /> <img src="image4.png" alt="Image 4" /></td>
</tr>
</tbody>
</table>
Recall: MNIST Digit Classification

Task specification:
- Input features: binary pixel values
- Output: a digit classification (0-9)

Issues with Naïve Bayes classifier:
- Can overfit to individual pixels
- Not robust to scaling, movement left/right, etc.
Convolutional Neural Networks

Image

Convolution
padding = 1,
kernel = 3x3,
stride = 1
+ ReLU

32 x 28 x 28

Max pooling
Kernel = 2x2,
Stride = 2

32 x 14 x 14

Convolution
padding = 1,
kernel = 3x3,
stride = 1
+ ReLU

64 x 14 x 14

Max pooling
Kernel = 2x2,
Stride = 2

64 x 7 x 7

Flatten

3136 x 128

128 x 10

0

1

9
Convolution in 1D

- Basic idea: define a new function by averaging over a sliding window
- Example in one dimension: smoothing
Convolution in 2D

- Filters in two dimensions: same idea but apply over a square patch of inputs (often 3x3 or 5x5)

- Applications:
  - Blurring
  - Sharpening
  - Feature detection
  - ...
**Convolutional Neural Networks**

- Key idea: learn the filter weights via backprop
Benchmarking on ImageNet

- **2010**: 28.2 (Lin et al)
- **2011**: 25.8 (Sanchez & Perronnin)
- **2012**: 16.4 (Krizhevsky et al (AlexNet))
- **2013**: 11.7 (Zeiler & Fergus)
- **2014**: 7.3 (Simonyan & Zisserman (VGG))
- **2014**: 6.7 (Szegedy et al (GoogLeNet))
- **2015**: 3.6 (He et al (ResNet))
- **2016**: 3 (Shao et al)
- **2017**: 2.3 (Hu et al (SENet))
- **Human**: 5.1

**First CNN-based winner**: 2012 (16.4)

- **152 layers**: 2014 (Szegoey et al (GoogLeNet)), 2015 (He et al (ResNet)), 2016 (Shao et al), 2017 (Hu et al (SENet))

- **19 layers**: 2013 (Zeiler & Fergus)
- **22 layers**: 2014 (Simonyan & Zisserman (VGG))
ResNet (He, et al. 2015)

**Key idea:**
- Want deeper networks with more parameters, but training signal becomes weak
- Add “skip” connections between layers so that there are shorter paths between early parameters and the final loss function

**ResNet:**
- 152-layer model for ImageNet
- Massive improvement over all previous CNN-based classification models circa 2015
Image Classification

cat
dog
horse
person
airplane
house
...

- cat
- dog
- horse
- person
- airplane
- house
- ...
a cat standing on a desk
Image Captioning with RNNs

**Input:** Image $I$

**Output:** Sequence $y = y_1, y_2, \ldots, y_T$

**Encoder:** $h_0 = f_W(z)$
where $z$ is spatial CNN features
$f_W(.)$ is an MLP

**Decoder:** $y_t = g_V(y_{t-1}, h_{t-1}, c)$
where context vector $c$ is often $c = h_0$

![Diagram showing the process of image captioning with RNNs](image)

- Extract spatial features from a pretrained CNN
- Features: $H \times W \times D$
- CNN
- MLP
- $z_{0,0}$, $z_{0,1}$, $z_{0,2}$, $z_{1,0}$, $z_{1,1}$, $z_{1,2}$, $z_{2,0}$, $z_{2,1}$, $z_{2,2}$
- $h_0$ connected to $c$
- $h_1$, $h_2$, $h_3$, $h_4$
- $y_0$, $y_1$, $y_2$, $y_3$, $y_4$
- [START] person wearing hat [END]

Image Captioning with RNNs + Attention

This entire process is differentiable.
- model chooses its own attention weights. No attention supervision is required

Extract spatial features from a pretrained CNN

Image Captioning with Transformers

Input: Image I
Output: Sequence $y = y_1, y_2, ..., y_T$

Encoder: $c = T_w(z)$
where $z$ is spatial CNN features
$T_w(.)$ is the transformer encoder

Decoder: $y_t = T_D(y_{0:t-1}, c)$
where $T_D(.)$ is the transformer decoder
Image Captioning with Vision Transformers

Need to learn these tokens
1. Learn good feature extractors from self-supervised pretext tasks, e.g., predicting image rotations.

2. Attach a shallow network on the feature extractor; train the shallow network on the target task with small amount of labeled data.

lots of unlabeled data

self-supervised learning

feature extractor (e.g., a convnet)

supervised learning

evaluate on the target task

small amount of labeled data on the target task

e.g. classification, detection

bird

conv

fc

90°
Contrastive Learning
Representation Learning: SimCLR

(a) Original  
(b) Crop and resize  
(c) Crop, resize (and flip)  
(d) Color distort. (drop)  
(e) Color distort. (jitter)  
(f) Rotate $\{90^\circ, 180^\circ, 270^\circ\}$  
(g) Cutout  
(h) Gaussian noise  
(i) Gaussian blur  
(j) Sobel filtering
Representation Learning: SimCLR

(a) Original  (b) Crop and resize  (c) Crop, resize (and flip)  (d) Color distort. (drop)  (e) Color distort. (jitter)

(f) Rotate \{90^\circ, 180^\circ, 270^\circ\}  (g) Cutout  (h) Gaussian noise  (i) Gaussian blur  (j) Sobel filtering
Autoencoders

An compressed low dimensional representation of the input.

Ideally they are identical.

\[ x \approx x' \]
Denoising Autoencoder

Original input $\mathbf{x}$ Partially destroyed input $\tilde{\mathbf{x}}$ Input

Encoder $g_\phi$ Bottleneck! Decoder $f_\theta$

Reconstructed input $\mathbf{x'}$

Ideally they are identical. $\mathbf{x} \approx \mathbf{x'}$

An compressed low dimensional representation of the input.
Generative Adversarial Networks

Idea: train a network to guess which images are real and which are fake!

"is this a real image"

This model can then serve as a loss function for the generator!
Diffusion Models

Forward diffusion process (fixed)

Data ———> Noise

Reverse denoising process (generative)
CLIP and DALL-E

1. Contrastive pre-training

pepper the aussie pup

2. Create dataset classifier from label text

plane

3. Use for zero-shot prediction

dog

bird

a photo of a (object).

...