From Perceptron to AI: The Rise of Large Language Models

Lecture 2 of 4:
How do LLMs work?

Dr. Michael Stobb
Thursday Forum
Coe College

February 2026

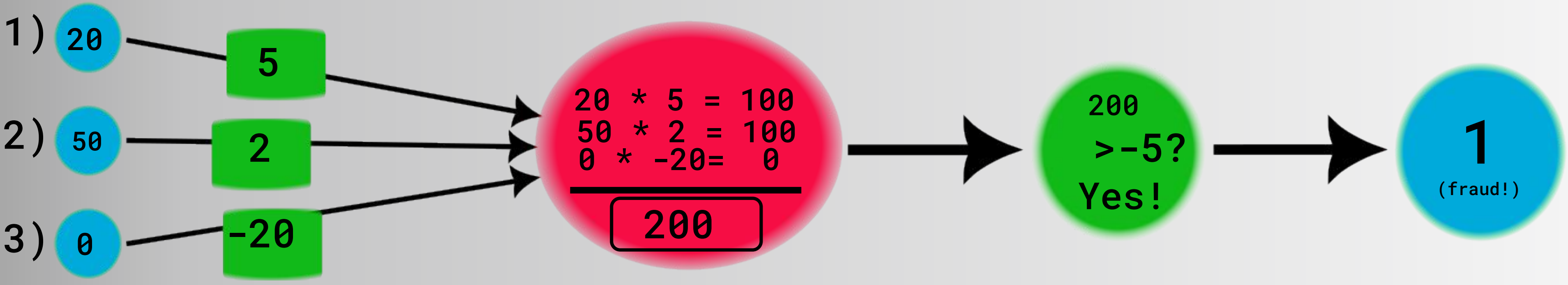
Last Time…

- Discussed “How did we get here?”
  - Environment was ready
  - OpenAI showed investors a predictable path to success

- Discussed the fundamental building block of AI:
  - An artificial neuron called a *Perceptron*
  - Inspired by real brain neurons
  - Accounts for over 60% of modern AI models

Recall a Perceptron Example

- Credit Card Fraud: Random internet purchase
  - Purchased $2000 laptop → 20
  - Purchased in Russia 5000 miles away → 50
  - Online purchase, so no chip → 0



Finding the Magic Numbers

- Finding the weights and threshold manually is hard
  - Basically impossible for larger problems

- Instead, we need an ***algorithm*** to learn them!
  - Turns out we can do this by looking at *examples*

- Imagine you have many examples that you **KNOW** are true

Known Input:

1. $5000
2. 3000 miles away
3. No chip used

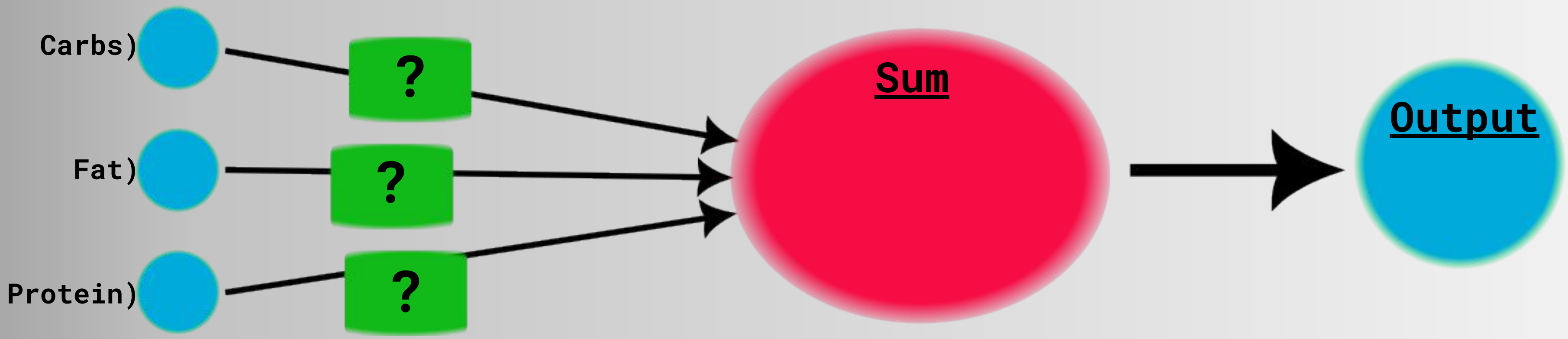
→

Known Output:

1. Definitely Fraud

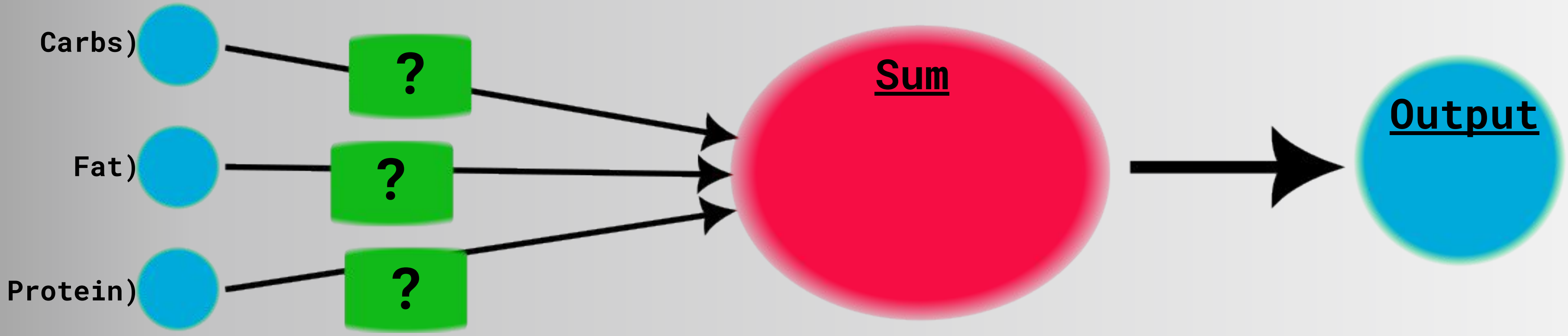
Finding the Magic Numbers

- Let's consider a *simpler* perceptron model:
  - This one will just predict total calories in food
  - Input will be:
    - Grams of Carbs
    - Grams of Fat
    - Grams of Protein
  - Output will be:
    - Total Calories



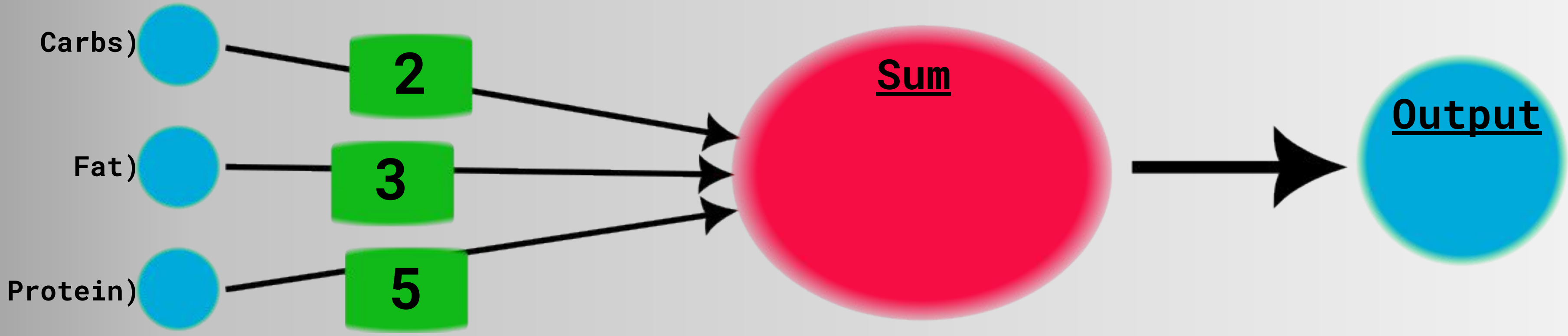
Finding the Magic Numbers

- Start by selecting RANDOM values for them



Finding the Magic Numbers

- Start by selecting RANDOM values for them
- Then test a KNOWN example
    - Let's try *Oreo Cookies*



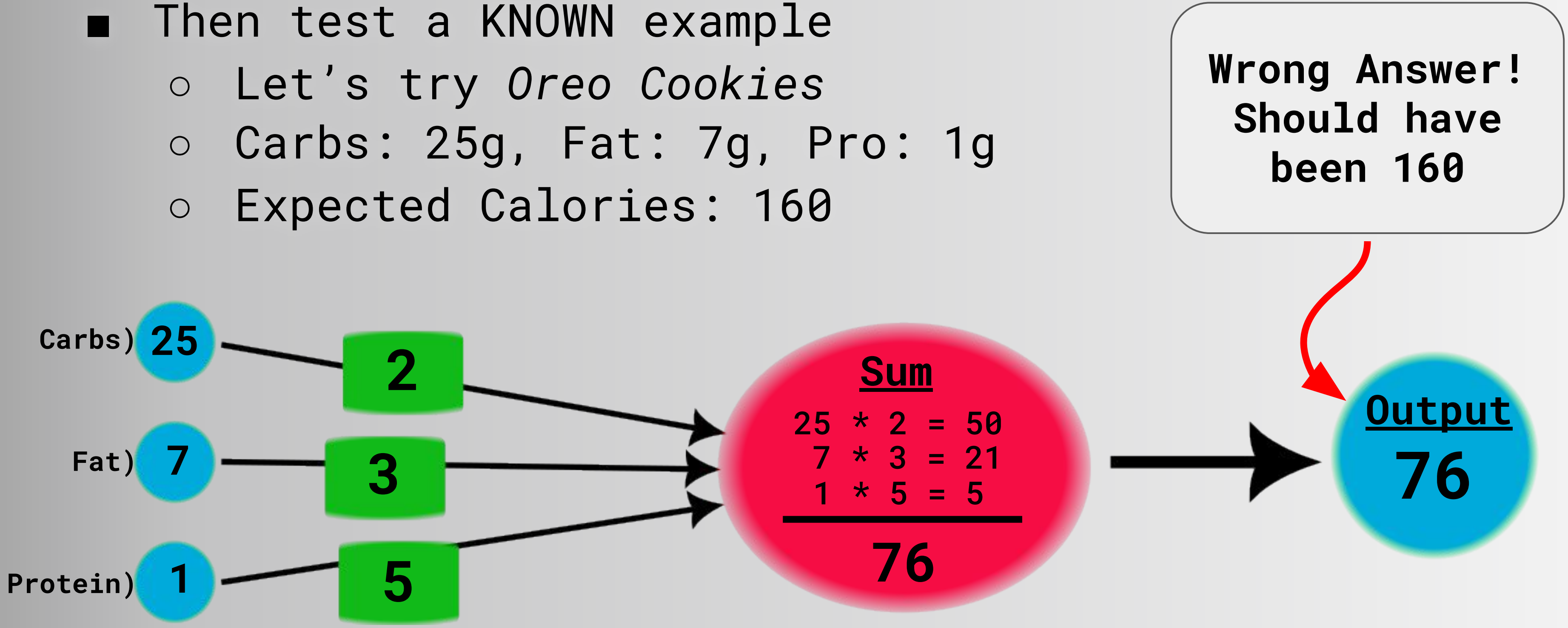
Finding the Magic Numbers

- Nutritional Information for Oreos:
  - Carbs: 25g
  - Fat: 7g
  - Protein: 1g

  - Total Calories: 160

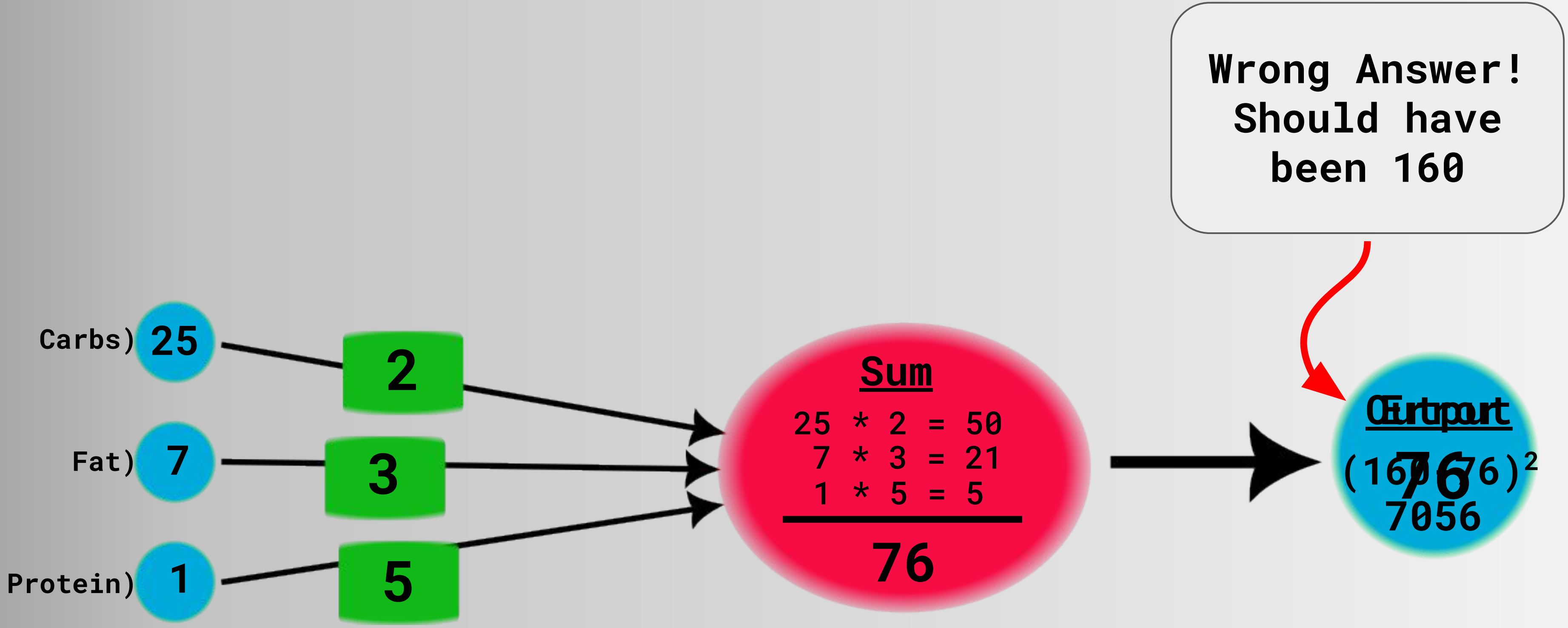
Finding the Magic Numbers

- Start by selecting RANDOM values for them
- Then test a KNOWN example
  - Let’s try *Oreo Cookies*
  - Carbs: 25g, Fat: 7g, Pro: 1g
  - Expected Calories: 160



Finding the Magic Numbers

- Use the "***Backpropagation***" algorithm



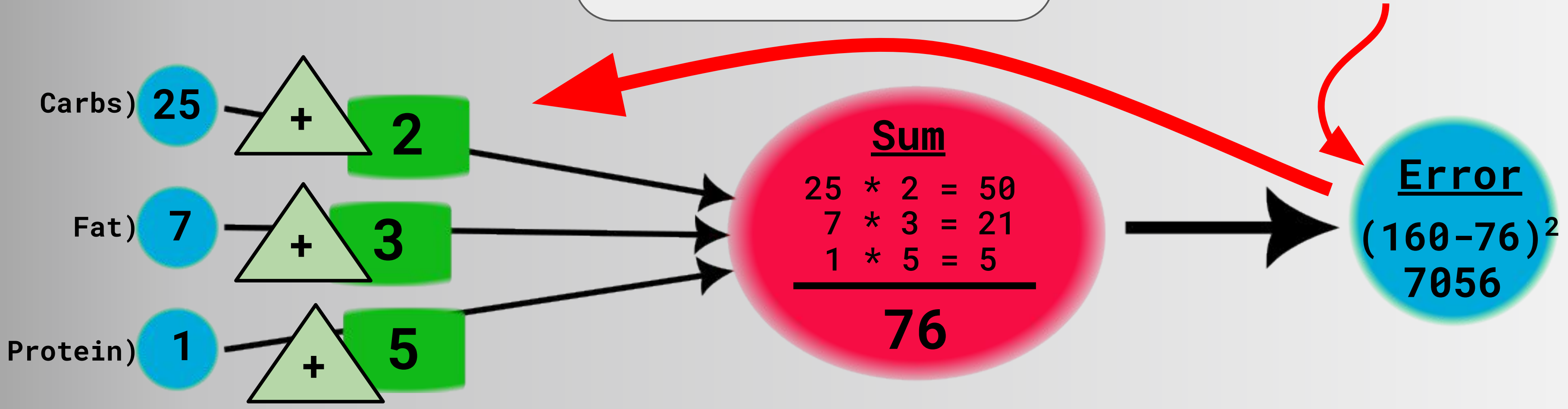
Finding the Magic Numbers

- Use the "*Backpropagation*" algorithm

We do the "Push" using *math*, specifically the *partial derivative*

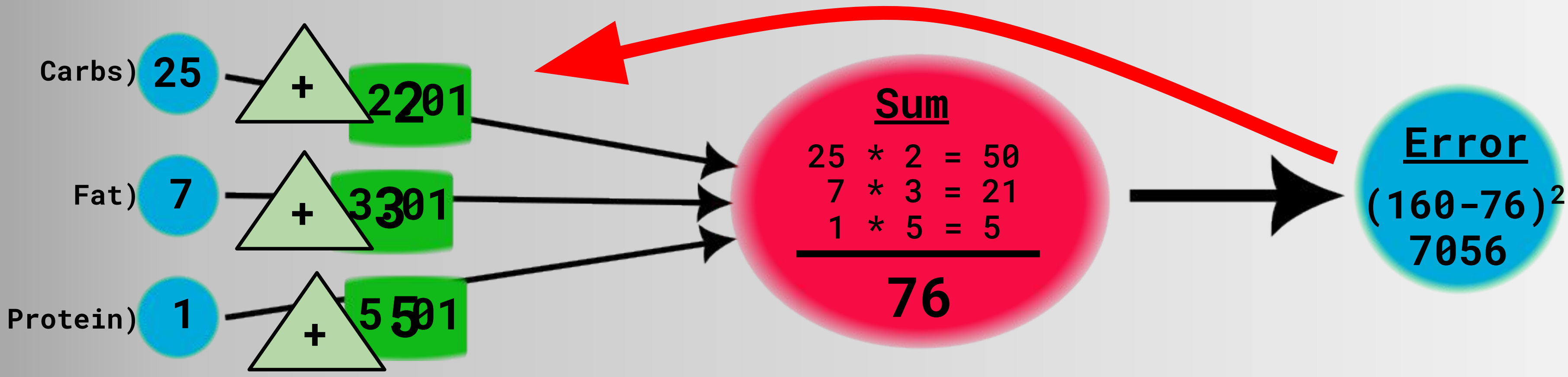
"Push" the error back to see where the GREEN numbers went wrong!

Wrong Answer! Should have been 160



Finding the Magic Numbers

- Use the "***Backpropagation***" algorithm
- Each example changes the weights by a TINY amount
- Repeat this process ***many*** times with new examples
- Each new example makes the numbers a bit "*better*"



Finding the Magic Numbers

- This "***Backpropagation***" algorithm allows the machine to "learn" from examples

- First invented in 1960s but not really used until the infamous 1986 paper by Geoffrey Hinton (and Rummelhart and Williams)

- Algorithm requires TONS of known examples to work

- All the complex math is easily done with computers
  - Same math as used in computer graphics!

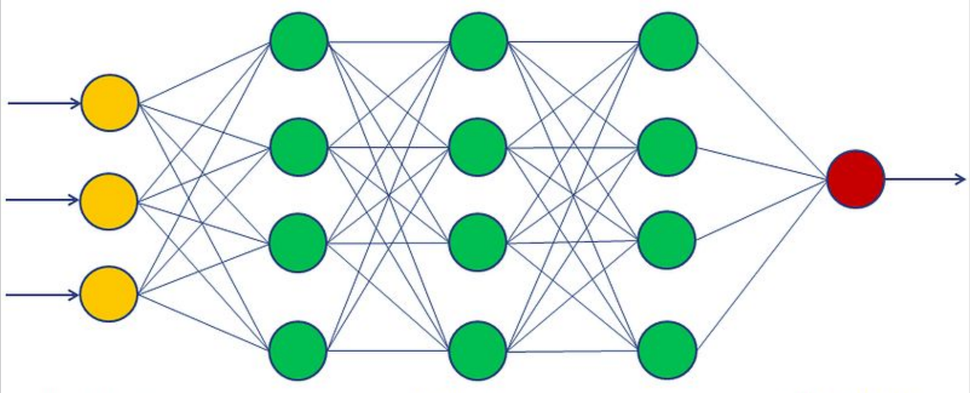
One Perceptron is NOT Enough

- All our examples have been *single* perceptrons

- But we discussed last time that one perceptron is incapable of capturing complex data

- We therefore introduced the

Multilayer Perceptron



One Perceptron is NOT Enough

- Multilayer Perceptrons *CAN* capture complex patterns

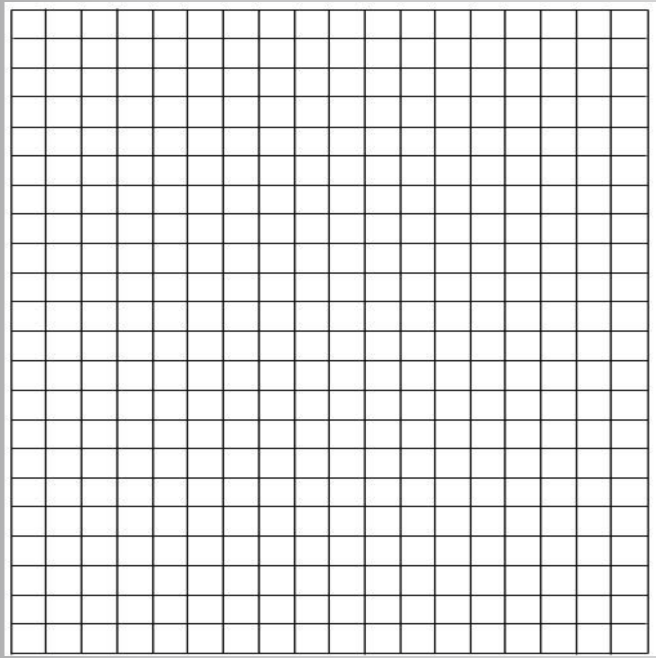
- They are just much harder to find their magic numbers
  - We call that process "training"
  - But it works the same as before:

    - Start with Random Values
    - Use examples to *nudge* your values better
    - Repeat until you have "good" numbers

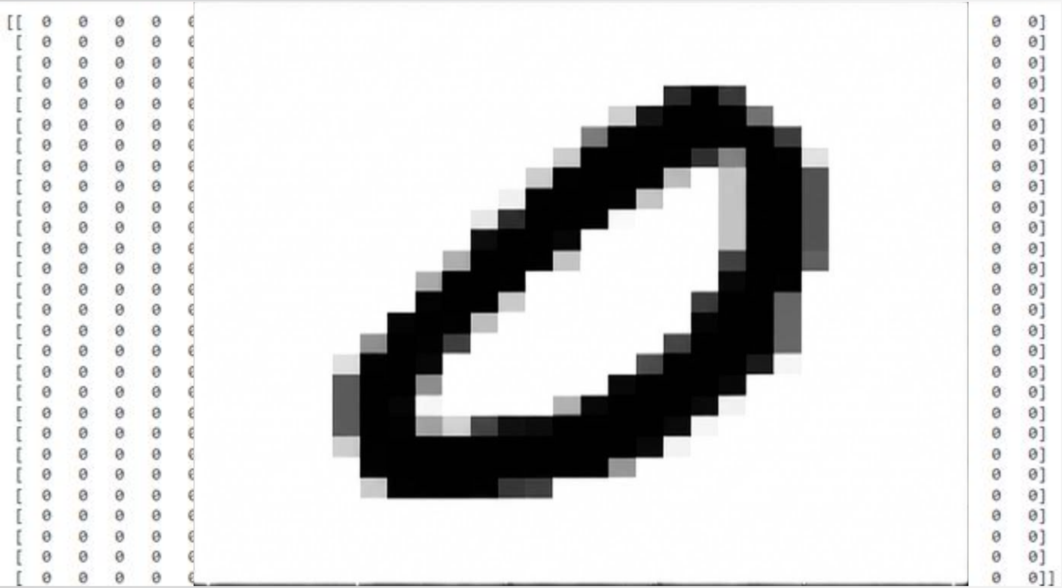
  - This process can take a LONG time

Multilayer Perceptron Example

- Imagine that instead of 3 inputs, you have a GRID:

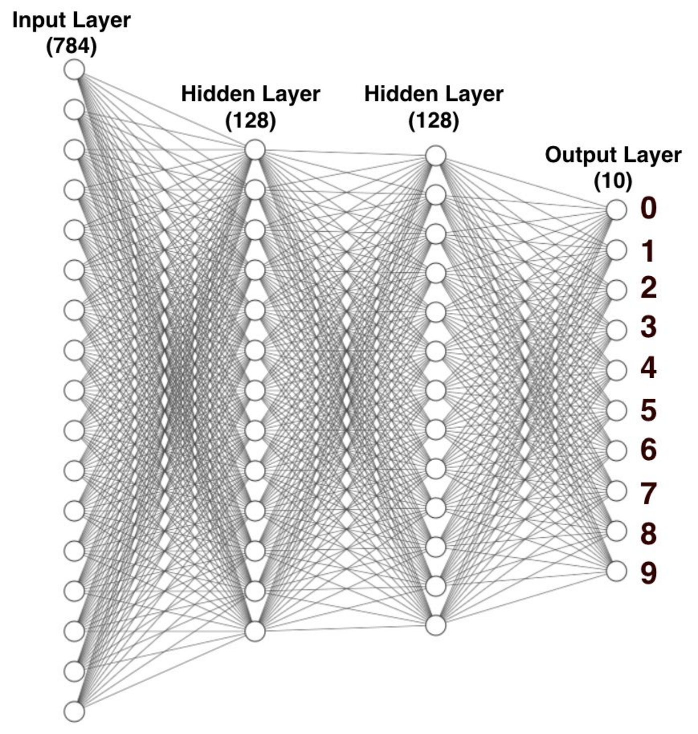


- Each grid square tells you how much INK goes there



Multilayer Perceptron Example

- The grid can be the INPUT for our network



- Finding magic numbers is hard, but still works!

How to Handle Words?

- All the examples we have looked at have used NUMBERS as both their *inputs* and *outputs*

- But we want to understand how ***Language Models*** work!

- Essentially each word (or word-part) is given a unique BARCODE

"apple"

"coffee"

"cake"

Now our model only needs to produce NUMBERS!

We can "look up" our words after

Transformer Based Neural Networks

- The Multilayer Perceptron is important
  - Accounts for 60-70% of modern AI models

- The other 30-40% comes from the ***Transformer***

- This is very particular *structure* of Neural Network
  - Was invented by Google Researchers in 2017
  - Presented in paper "*Attention is all you Need*"
  - Named that because it *transforms* inputs into outputs

- Instead of examining one word at a time, it examines ALL the words at the SAME time
  - Able to determine *how* words relate to one another

Transformer Based Neural Networks

- Example Sentence:

"The river was beautiful. My favorite area was the bank by the large oak tree."

- Question: What does the word "bank" here refer to?
  - It's a *riverbank*, not the bank used for money

- We only know that by looking at the words around it
  - Transformers understand *context* between words

- Transformer models can "read" all their input at once

Large Language Models

- Large Language Models (LLMs) are just one particular example of Transformer Based Neural Networks

  - They take a series of words as input
  - They give out a single word as output

- “Large” here is a bit misleading
  - Smallest LLMs are about 1GB (fit on phone)
  - Largest LLMs are multiple TB (full server rack)

Big Idea: All LLMs work by taking a group of words as input and predicting the single ***next*** word.

Large Language Models

Example:

- Predict the next word:

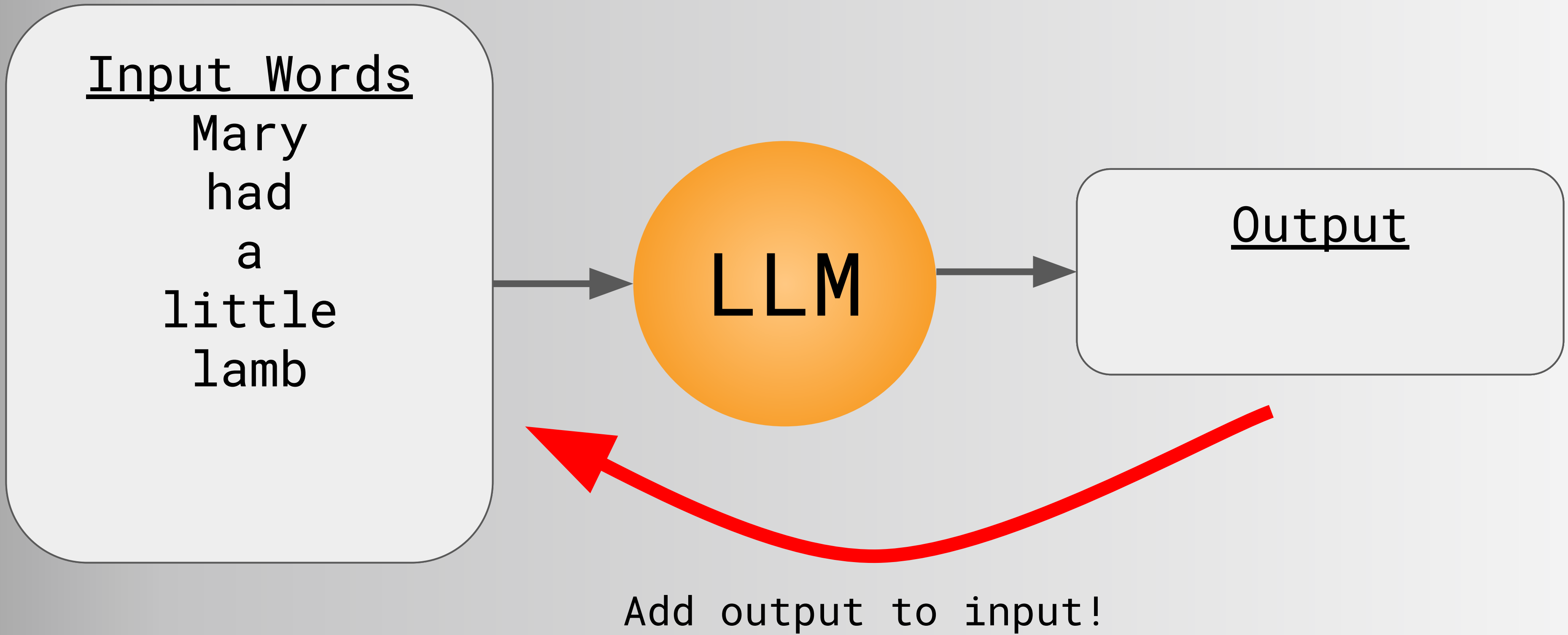
  Mary had a little lamb

- How did we all know that?
  - We have seen/heard/read prior examples

- This is basically that an LLM does as well!
  - Based on the many examples it has seen, it tries to predict the next word

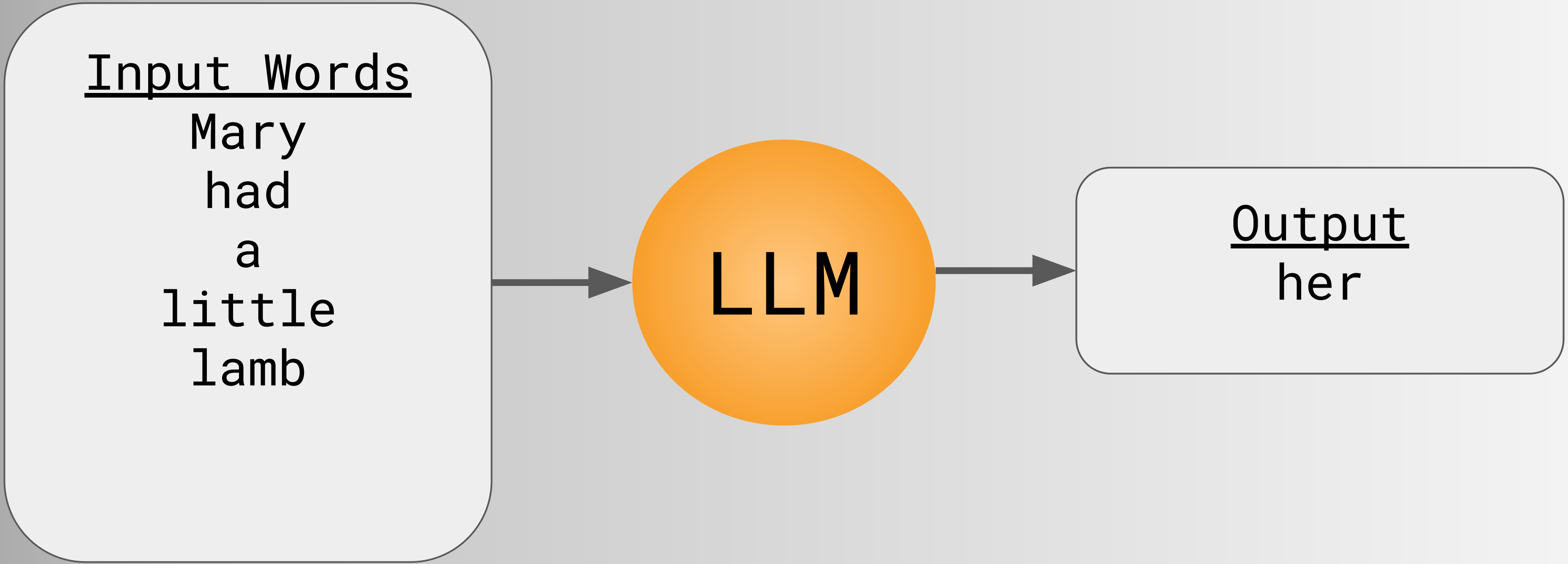
Large Language Models

Run Number: 1



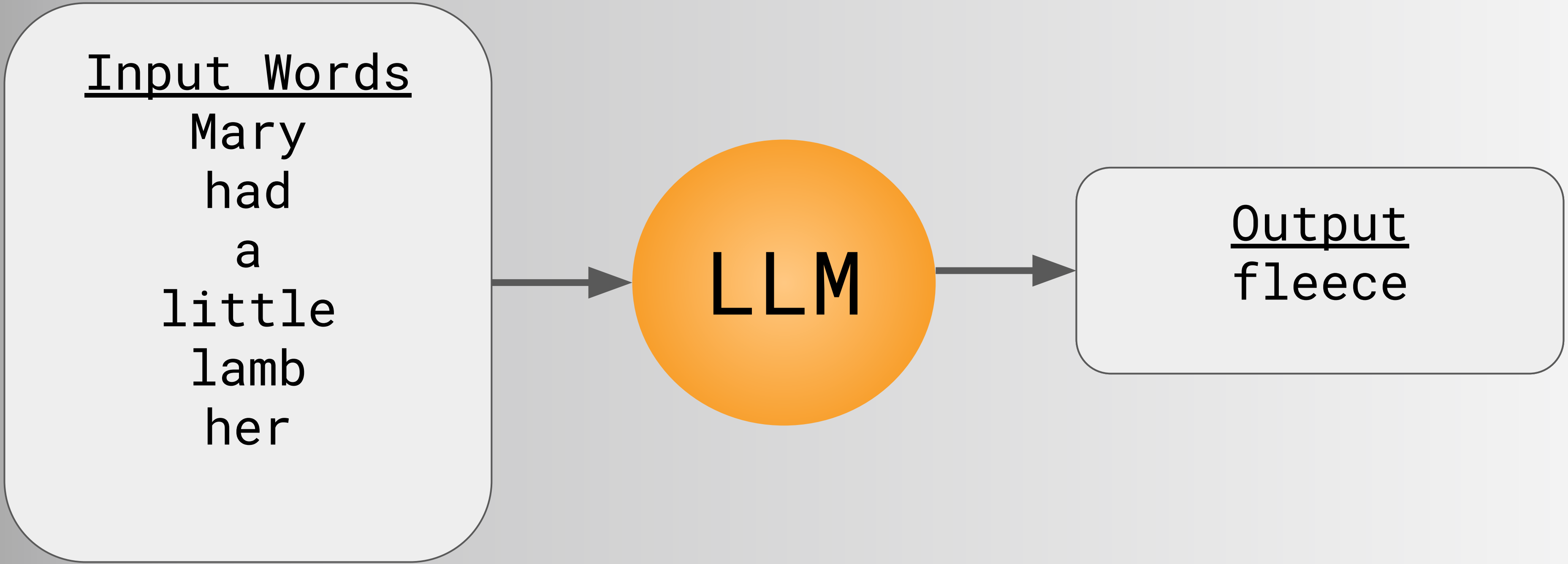
Large Language Models

Run Number: **2**



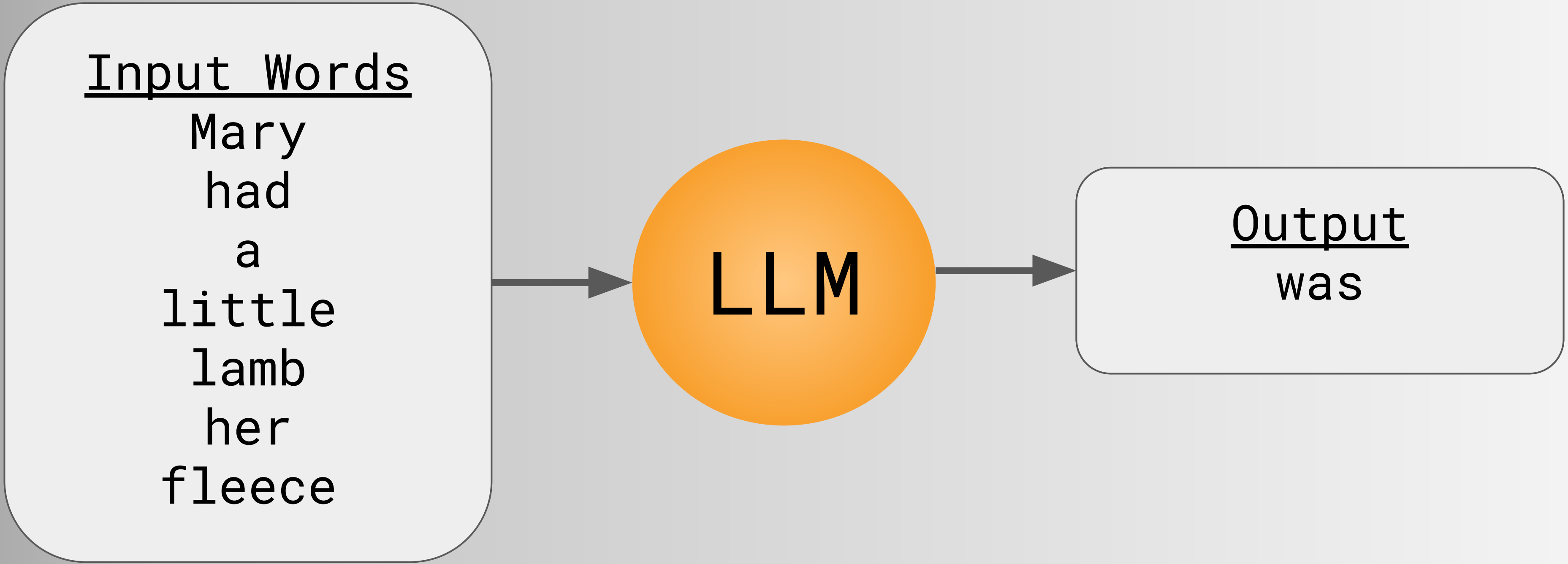
Large Language Models

Run Number: **3**



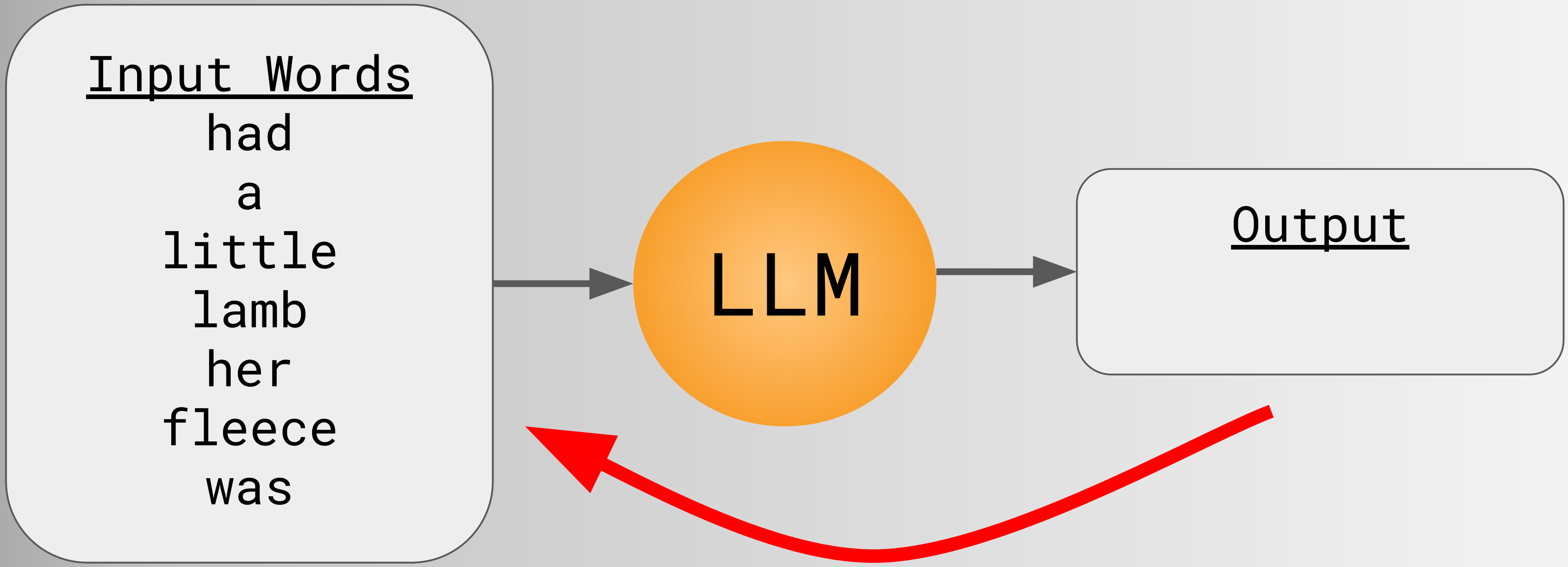
Large Language Models

Run Number: **4**



Large Language Models

Run Number: **4**



Input is now *full!*

Delete the earliest word!

Large Language Models

Run Number: **5**

Input Words

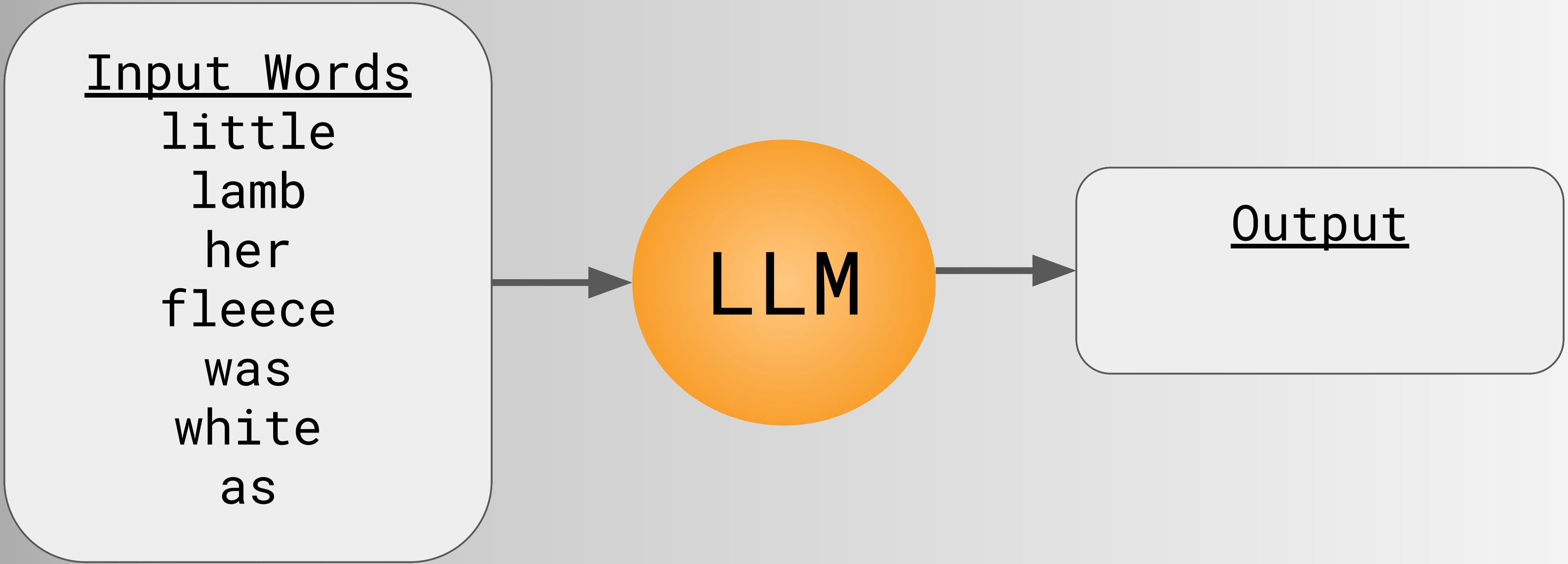
a
little
lamb
her
fleece
was
white

LLM

Output

Large Language Models

Run Number: **6**



Large Language Models

Run Number: 7

Input Words
little
lamb
her
fleece
was
white
as

LLM

Output
snow

Check out this app
for more [predictions]

Large Language Models and Data

- Training these models requires tons of *KNOWN* examples
- Where do these examples come from?
  - These are *constructed* from the data taken from us!

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness…

-Charles Dickens

<u>Input</u>	<u>Output</u>
It was the best of	times

Large Language Models and Data

- Training these models requires tons of *KNOWN* examples

- Where do these examples come from?
  - These are *constructed* from the data taken from us!

> It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness…
>
> -Charles Dickens

Input	Output
It was the best of	times
was the best of times	it

Large Language Models and Data

- Training these models requires tons of *KNOWN* examples

- Where do these examples come from?
  - These are *constructed* from the data taken from us!

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness…
-Charles Dickens

Input	Output
It was the best of	times
was the best of times	it
the best of times it	was

Large Language Models and Data

- Training these models requires tons of *KNOWN* examples

- Where do these examples come from?
  - These are *constructed* from the data taken from us!

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness…

-Charles Dickens

Input	Output
It was the best of	times
was the best of times	it
the best of times it	was
best of times it was	the

Large Language Models and Data

- Training these models requires tons of *KNOWN* examples

- Where do these examples come from?
  - These are *constructed* from the data taken from us!

It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness, it was the epoch of belief, it was the epoch of incredulity, it was the season of Light, it was the season of Darkness…

-Charles Dickens

Input	Output
It was the best of	times
was the best of times	it
the best of times it	was
best of times it was	the
of times it was the	worst

LLM Recap

1. Perceptrons alone are not enough, we need many of them linked

2. Neural Networks require training (finding the magic numbers), which can be done using *Backpropagation*

3. Backpropagation requires *known* examples of input and output

4. Large Language Models (LLMs) are a combination of Perceptrons and Transformers, which allow the model to understand context

5. Words are stored as unique numbers (or barcodes)

6. LLMs take a series of words and predict the *next* word

7. Example data is created from all human written data

Training a LLM

- We know LLMs require large amounts of data to train, but that is not all

- Modern AI models use THREE main phases for training:

1. **Pre-Training**: The model learns basic language
2. **Supervised Fine-Tuning**: The model learns to obey
3. **Reinforcement Learning**: The model learns to be *nice*

1) <u>Pre-Training</u>

- As the name implies, this is really the step *before* training a model

- Starts with a blank, random model, with billions of *to-be-determined* numbers

- The model looks at billions of examples of language and learns:
  - Grammar, syntax, word meaning, punctuation, etc.

- If we stopped here, the model would be ***terrible***:
  - Simply a statistical parrot
  - Nothing but a very fancy “autocomplete”
  - Racist, vulgar, and quite rude

2) <u>Supervised Fine-Tuning</u>

- Starts with the finished pre-trained model
  - Knows basic grammar and language

- Now the model learns how to follow instructions
  - Back and forth conversations, rule following, template adherence, etc.

- The model learns from thousands of ***carefully*** made and curated conversations
  - These are highly proprietary for the individual labs

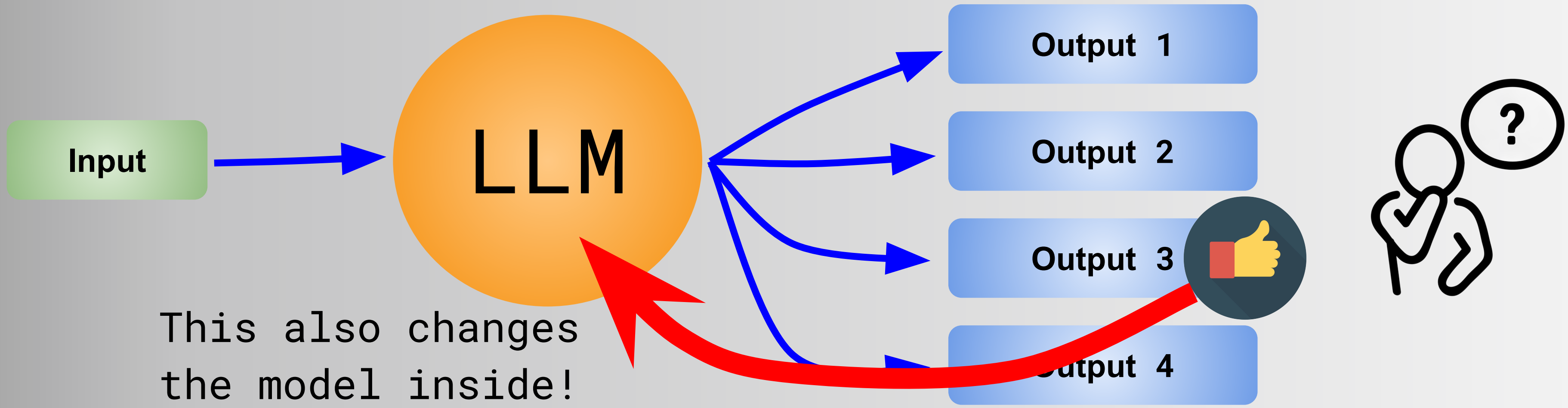
- The model itself is changed! The magic numbers are altered!
  - The model is *no-longer* ***just*** a next word predictor based on word frequencies
  - It is now trying to predict the "right" word next

3) Reinforcement Learning (with human feedback)

- Starts with the finished "fine-tuned" model
  - Knows grammar and language
  - Can follow basic instructions

- Now trained to be "nice", "polite", and "kind"
  - Very subjective, was a hard thing to figure out

RLHF



This also changes the model inside!

Model Training

- No one step is enough on its own to make a “good” model

- Must be done sequentially (one step after another), which takes extra time

- At the end, we hopefully have a model that is

    - Smart
    - Rule Following
    - Well Adjusted

- More training steps are being added!
    - Ex: Reinforcement learning on “Chains of Thought”

Measuring Success

- We have built a model
  - It’s been pre-trained, fine-tuned, and made “nice”

- Big Question:

How do we know the model is any good?

- Answer:

Benchmarks

- Benchmarks are standardized tests used to assess and compare AI performance on specific tasks

Measuring Success

- We need a ruler if we want to measure progress or to compare models

Goodhart's Law: When a measure becomes a target, it ceases to be a good measure.

- There are some specific problems for AI:
  - **Benchmaxing**: Model builders can (intentionally or otherwise) train their models for specific tasks
  - **Contamination**: If the example questions are published to the internet, they could be in the pre-training data
  - **Saturation**: Models can become saturated and too easy for models

MMMLU

- Multilingual Massive Multitask Language Understanding

- A multiple choice test administered in 14 different languages covering 57 different topics
  - Elementary math, US History, Law, Anatomy, etc.
  - About 16,000 questions in total

When you drop a ball from rest it accelerates downward at 9.8 m/s^2. If you instead throw it downward assuming no air resistance its acceleration immediately after leaving your hand is

(A) 9.8 m/s^2

(B) more than 9.8 m/s^2

(C) less than 9.8 m/s^2

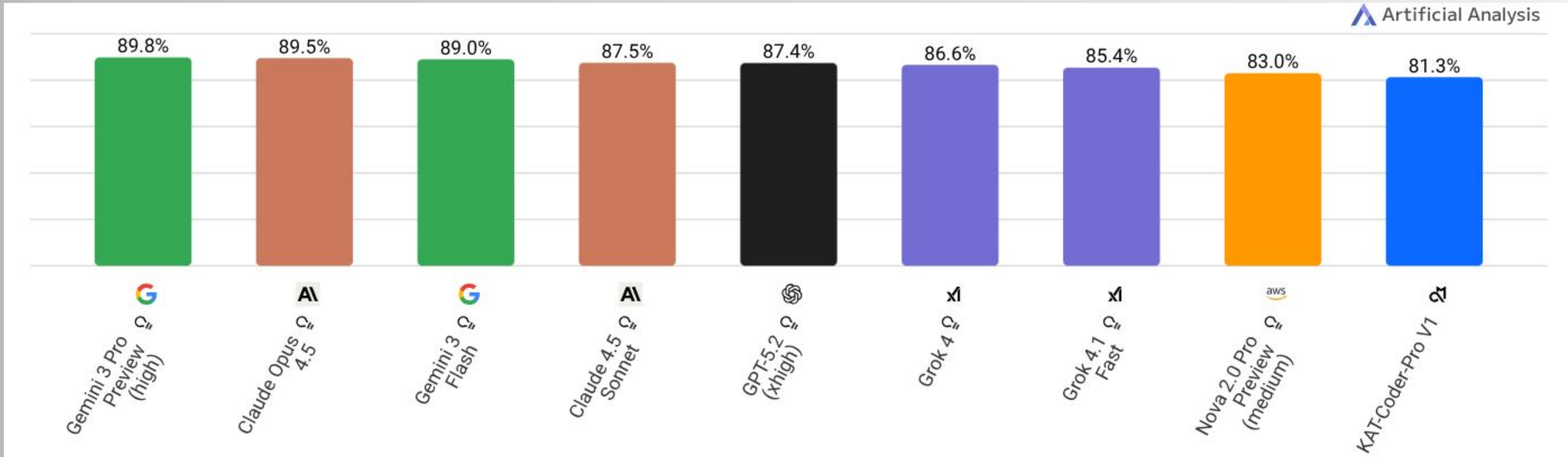
(D) Cannot say unless the speed of throw is given.

MMMLU

- Randomly guessing gets 25% on the exam (1 out of 4)

- Average human gets about 35% on the test

- An expert in their field would get approximately 90%
  - But only in their subject portion!



<u>Humanity's Last Exam</u>

- Designed as a successor to the MMMLU since models were routinely getting around 90% on it

- 2,500 total questions covering dozens of topics
  - Combination of *<u>free response</u>* and multiple choice

- Questions are PhD level, submitted by experts
  - Answers are NOT available via simple online searches

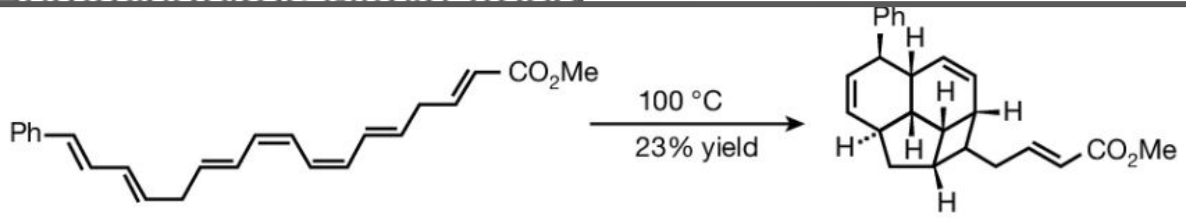
- Average human score would be very low, less than 1%
  - Expert score approximately 90% (only in their field)

Humanity's Last Exam

Hummingbirds within Apod

I am providing the standardized Biblical Hebrew source text from the
ms 104:7). Your task is to
n syllables. Please identify and
onsonant sound) based on the
unciation tradition of Biblical
y Khan, Aaron D. Hornkohl, Kim
dieval sources, such as the
ave enabled modern researchers
ts of Biblical Hebrew
n, including the qualities and
tters were pronounced as

(Psalms 104:7) ?



endiandric acid B methyl ester

The reaction shown is a thermal pericyclic cascade that converts the starting heptaene into endiandric acid B methyl ester. The cascade involves three steps: two electrocyclizations followed by a cycloaddition. What types of electrocyclizations are involved in step 1 and step 2, and what type of cycloaddition is involved in step 3?

Provide your answer for the electrocyclizations in the form of [nπ]-con or [nπ]-dis (where n is the number of π electrons involved, and whether it is conrotatory or disrotatory), and your answer for the cycloaddition in the form of [m+n] (where m and n are the number of atoms on each component).

Humanity's Last Exam

ARC-AGI-1

- Abstraction and Reasoning Corpus for Artificial General Intelligence 1

- Designed to test machine reasoning directly
  - Does not rely upon text or words at all
  - Instead, uses small visual puzzles

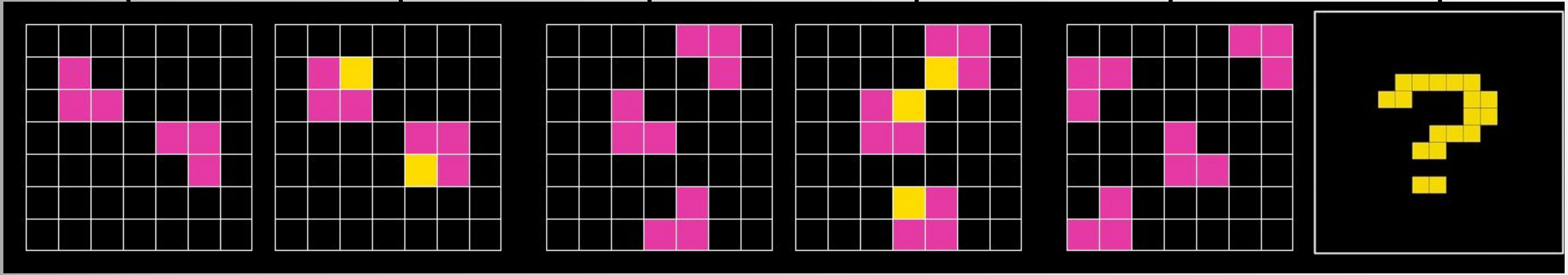
- About 1000 total questions, but with infinite possibilities

- Usually very easy for a human to complete (over 90%)

ARC-AGI-1

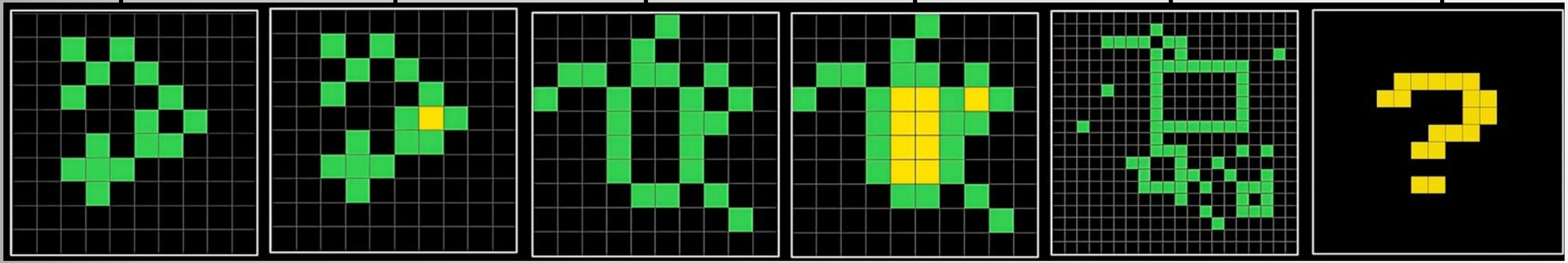
- **Sample 1**

Input Output Input Output Input Output

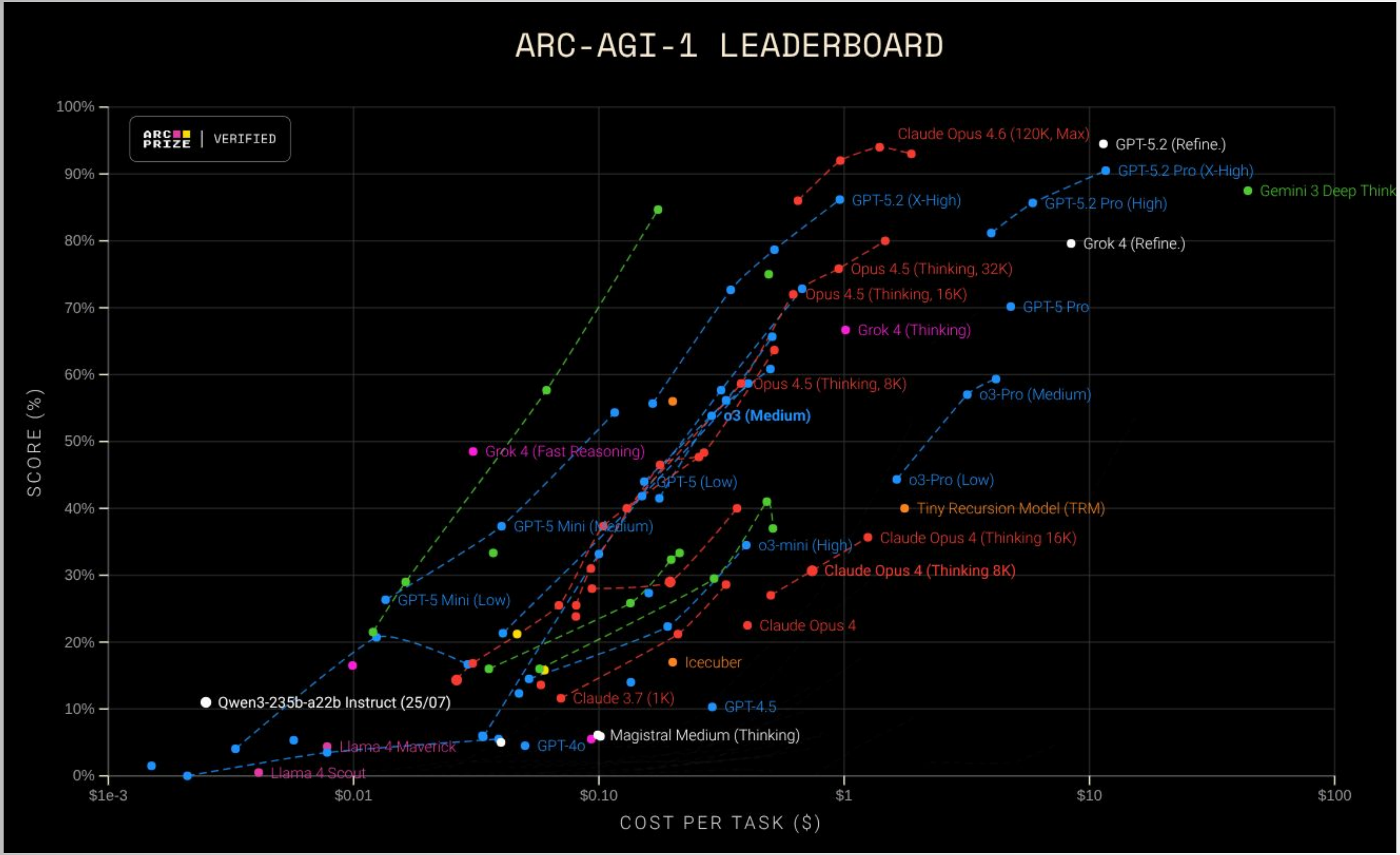


- **Sample 2**

Input Output Input Output Input Output



ARC-AGI-1


ARC-AGI-1 LEADERBOARD
ARC PRIZE | VERIFIED
SCORE (%)
COST PER TASK ($)
100%
90%
80%
70%
60%
50%
40%
30%
20%
10%
0%
$1e-3
$0.01
$0.10
$1
$10
$100
Claude Opus 4.6 (120K, Max)
GPT-5.2 (Refine.)
GPT-5.2 Pro (X-High)
Gemini 3 Deep Think
GPT-5.2 (X-High)
GPT-5.2 Pro (High)
Grok 4 (Refine.)
Opus 4.5 (Thinking, 32K)
Opus 4.5 (Thinking, 16K)
GPT-5 Pro
Grok 4 (Thinking)
Opus 4.5 (Thinking, 8K)
o3-Pro (Medium)
o3 (Medium)
Grok 4 (Fast Reasoning)
GPT-5 (Low)
o3-Pro (Low)
Tiny Recursion Model (TRM)
GPT-5 Mini (Medium)
o3-mini (High)
Claude Opus 4 (Thinking 16K)
Claude Opus 4 (Thinking 8K)
GPT-5 Mini (Low)
Claude Opus 4
Icecuber
Qwen3-235b-a22b Instruct (25/07)
Claude 3.7 (1K)
GPT-4.5
Magistral Medium (Thinking)
Llama 4 Maverick
GPT-4o
Llama 4 Scout

ARC-AGI-1

- Had reached human levels of competence as of 1 year ago
  - Though the *cost* to do it can be quite large

- Designers thought it would take many years to break
  - Instead, took less than two years

- They have since made an ARC-AGI-**2** benchmark
  - This is essentially a harder version of the first
  - Humans can still do it, but it takes longer
  - Models are currently getting over 70% on it

What does all this tell us?

- New Nature article (this month!) summarized recent AI benchmark results

- They concluded that modern AI models have reached the level of “*Artificial General Intelligence*”

  - “Insofar as individual humans have general intelligence, current LLMs do, too”

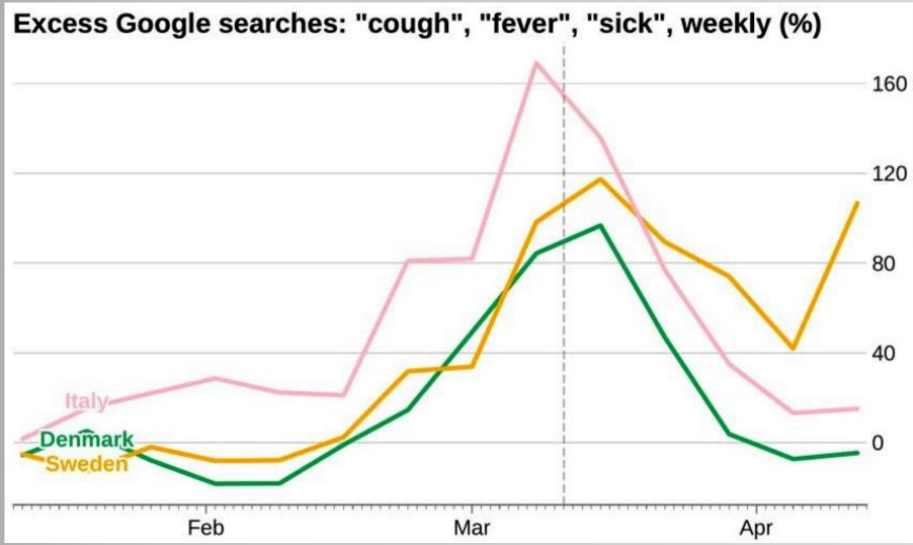
- AI researchers believe that progress will continue and models will continue to improve

- Regardless, current models are already very smart

CharXiv

- A visual test about charts and graphs

- The model is supplied a chart (visually) and then asked questions about that chart



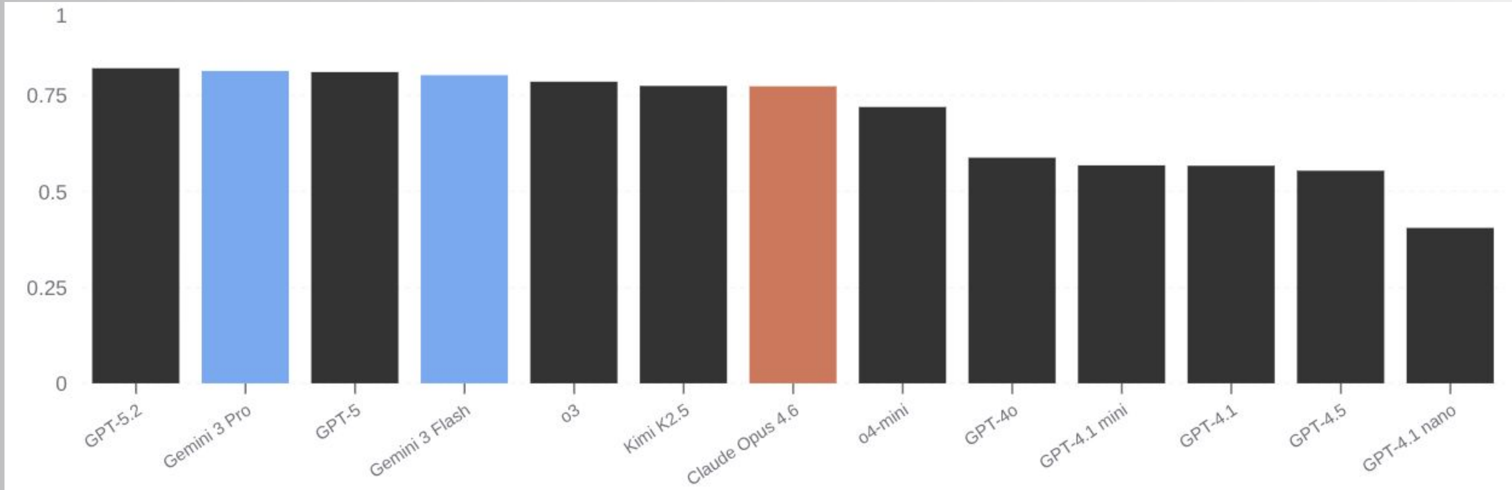
What is the name of the country that has a significant bounce for Excess Google searches of cough, fever and sick shortly after April?

CharXiv

- Has over 10,000 questions and over 2300 charts

- Minimum score is 0, since not multiple choice
  - Average human gets 80.5% on the reasoning test

- Modern models are getting better than human scores



Next Time…

- We will explore the impacts of AI!

Thank you!

For notes, further readings, and a full copy of the slides, just scan the QR code:

stobb.org/thursday_forum/2026_lecture_02/