JOHNS HOPKINS
CAREY BUSINESS SCHOOL

# Lecture 2

## BU.330.760 Generative AI for Business

Minghong Xu, PhD.
Associate Professor

# Recent News

Create a photo for this two figure play badminton in space on two sides of earth, use Pixar cg 3d style

Finishing touches

GEMINI

## Data Science Agent in Colab: The future of data analysis with Gemini

MAR 03, 2025

Jane Fine, Senior Product Manager
Mahi Kolla, Associate Product Manager
Ilai Soloducho, Senior Technical Program Manager

Share

## Here's how the Data Science Agent works:

1. **Start fresh:** Open a blank Colab notebook.

2. **Add your data:** Upload your data file.

3. **Describe your goals:** Describe what kind of analysis or prototype you want to build in the Gemini side panel (e.g., "Visualize trends," "Build and optimize prediction model", “Fill-in missing values”, “Select the best statistical technique”).

4. **Watch the Data Science Agent get to work:** Sit back and watch as the necessary code, import libraries, and analysis is generated in a working Colab notebook.

# Today's Agenda

- Encoder-Decoder Architecture
- Attention Mechanism and Transformers
- Generative AI Ecosystem

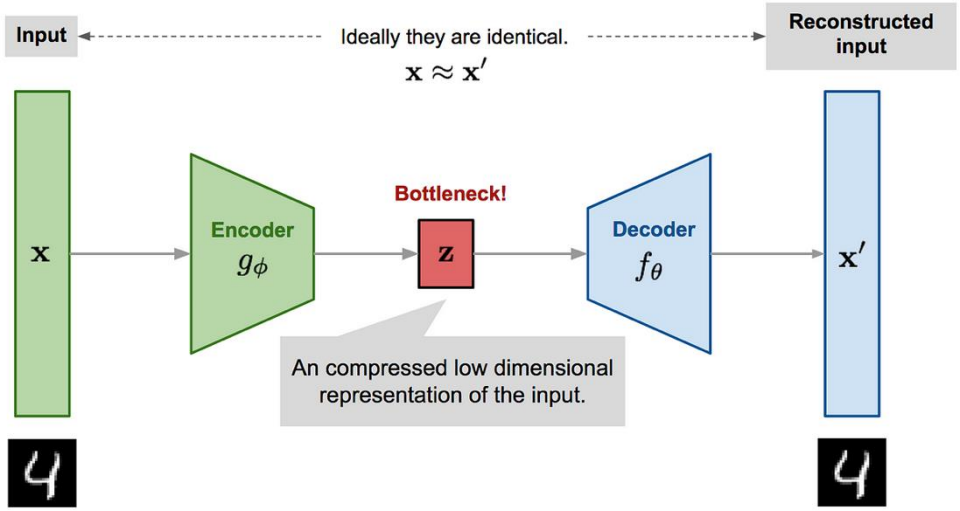
# Model and Architecture

- At the core of any generative AI system is the **model**
  - Mathematical representation of the system
- **Foundation model:** a model trained on board data that can be adapted to a wide range of downstream tasks

- **Architecture**: the structure of the model
  - Organization of parameters in an artificial neural networks
  - Generate the outputs

# Encoder-Decoder

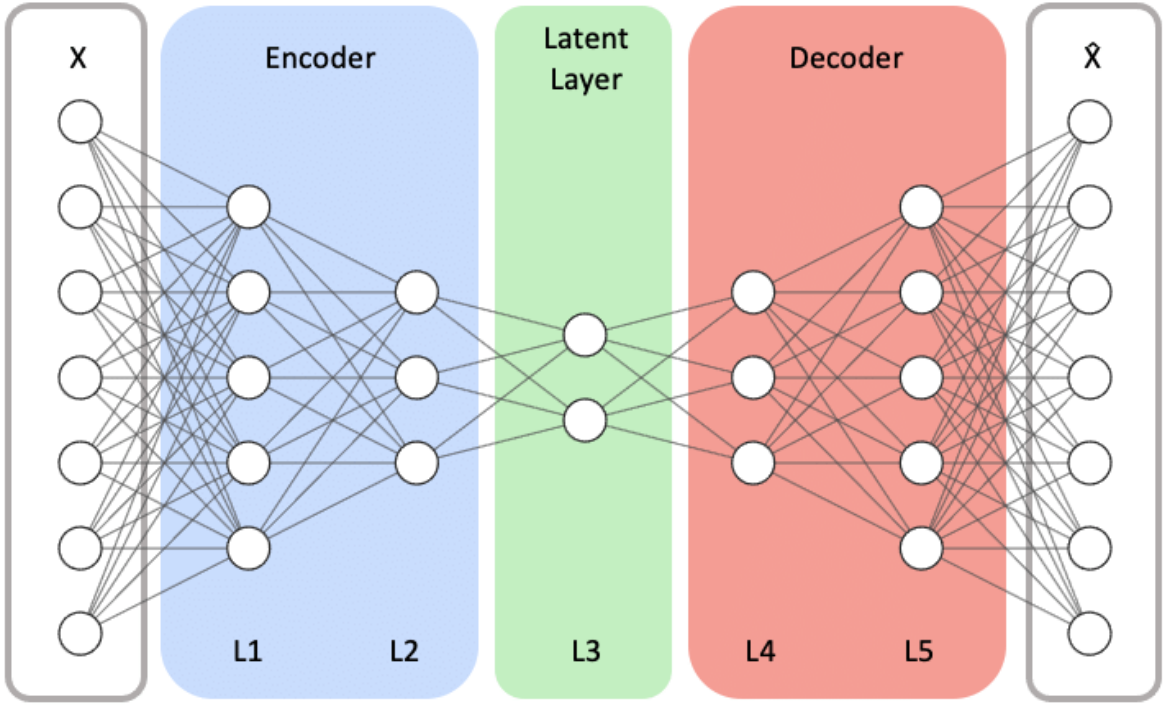
- A neural network architecture made up of two parts
- An encoder network: compress high-dimensional input data into a lower-dimensional embedding vector
- A decoder network: decompresses a given embedding vector back to the original domain
- Also called **autoencoder**
- *Have you played this game?*



# Encoder also performs Embedding

- The encoder attempts to embed as much information into the representation as possible, so that the decoder can reproduce an accurate reconstruction

- *Another embedding?*



# Token and Tokenizer (Recap)

- Break a stream of text into meaningful units
- Tokens: words, phrases, symbols
- https://platform.openai.com/tokenizer

- GPT-3: 500 billion tokens (about 300 billion words)

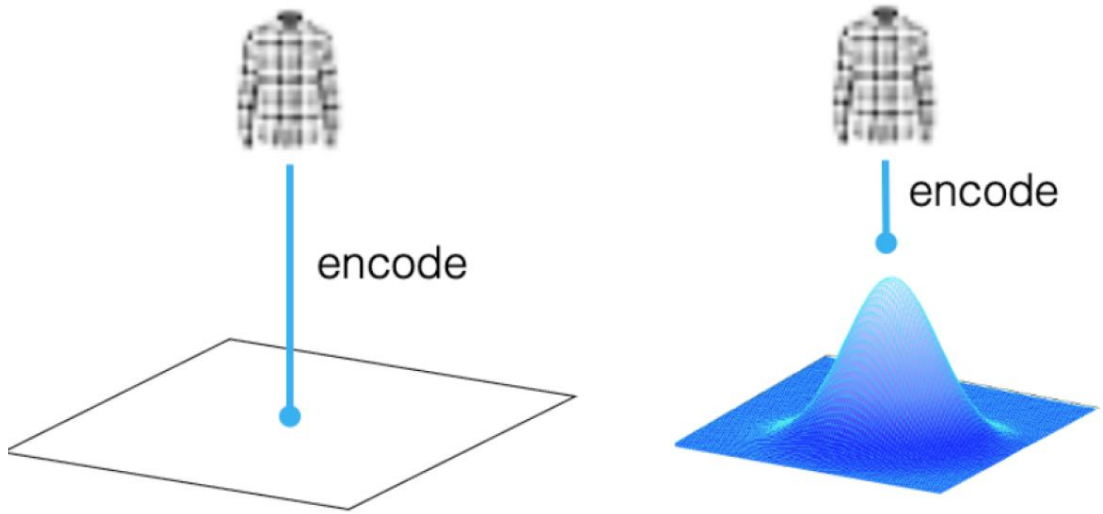
# Word Embedding (Recap)

- *Check the Token IDs on https://platform.openai.com/tokenizer*
- *Can we do better to represent each word?*
- Yes! We need the meaning
- Word Embedding: represent words in a dense vector format
  - Capture semantic meaning
  - Words that are semantically similar are close to each other in the vector space

- https://projector.tensorflow.org/

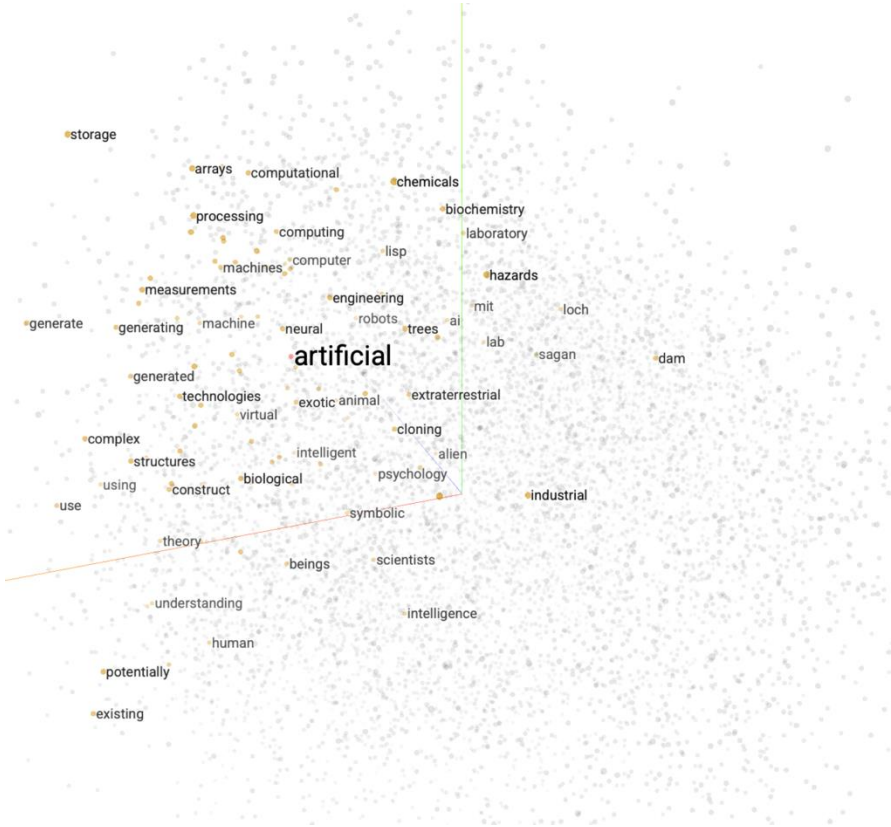
# Autoencoder vs VAE

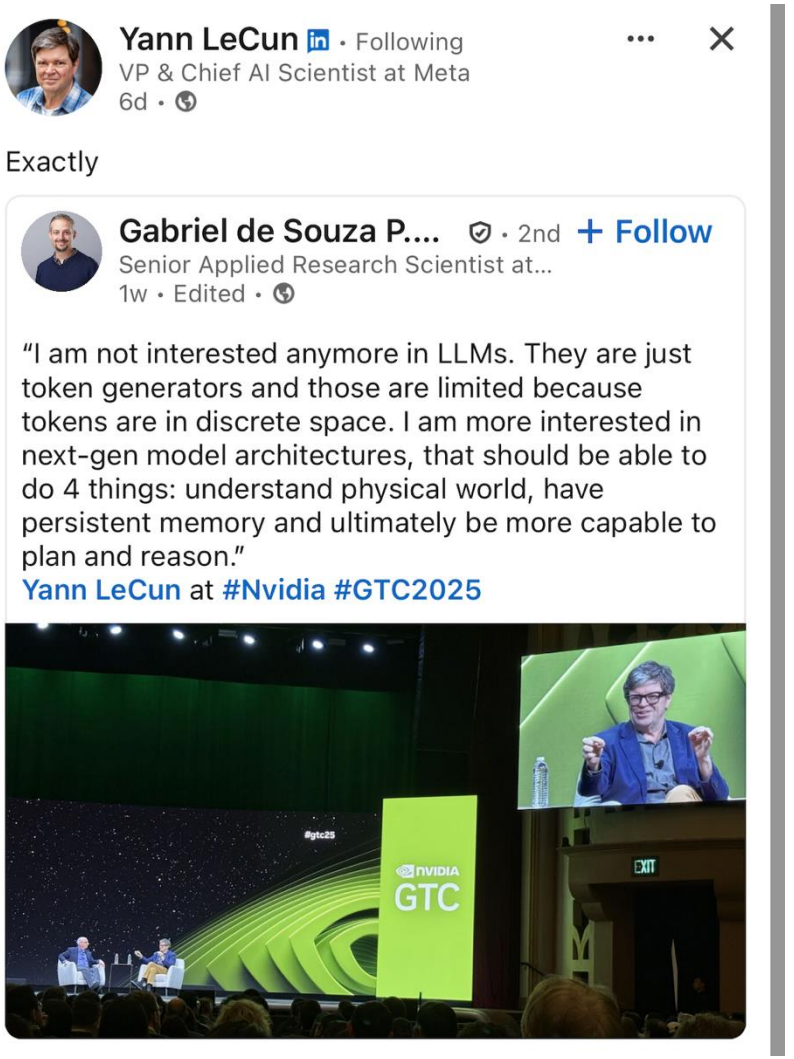
- Autoencoder: each data point is mapped directly to one point in the latent space
- Variational autoencoder: each data point is mapped to a multivariate normal distribution around a point in the latent space
- Decoder will be the same



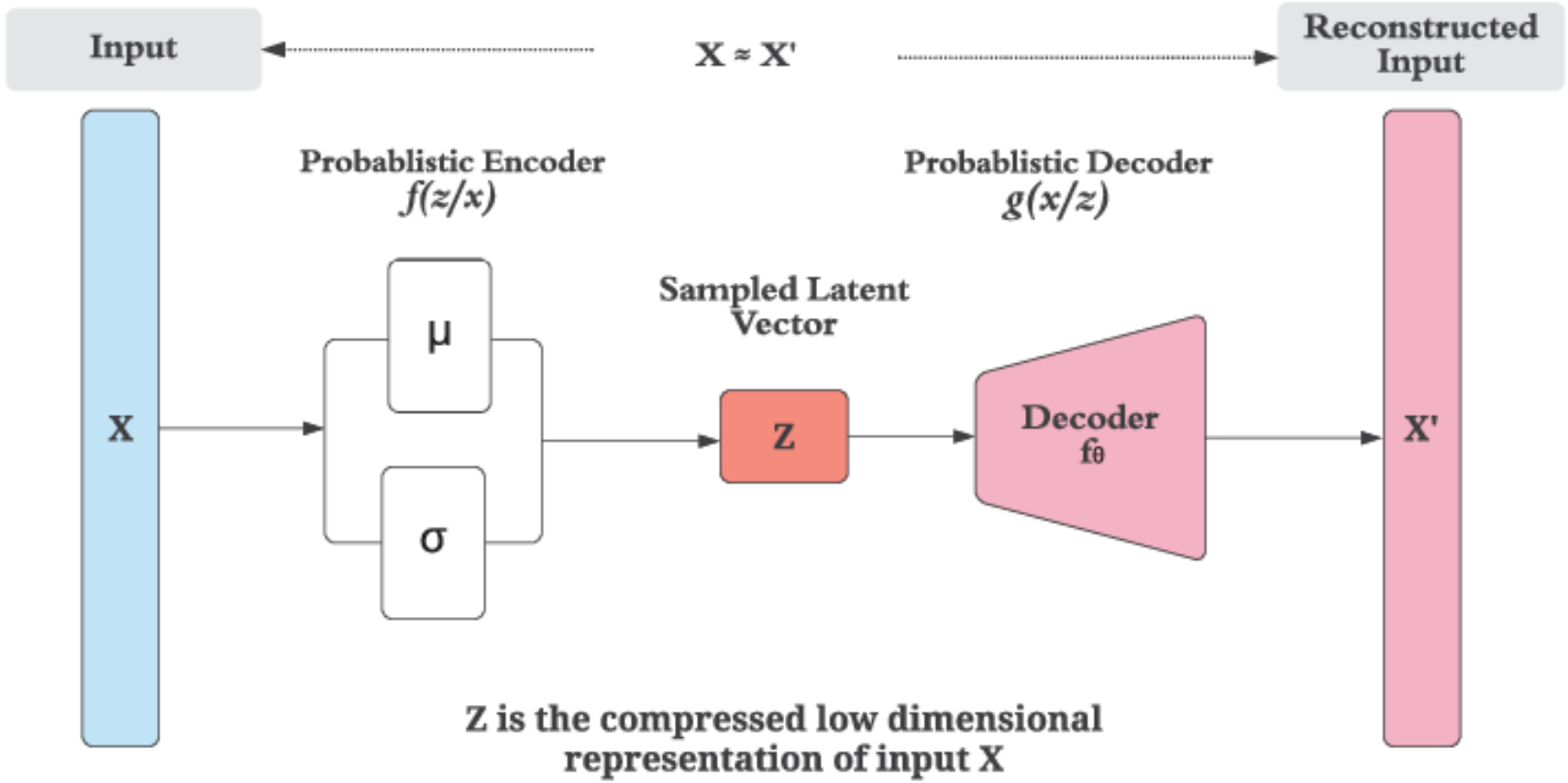
# Why Continuous

- https://projector.tensorflow.org/



# VAE Architecture (Optional)



# Transformer

- Introduced in 2017 by Google
- Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).
- An encoder-decoder architecture with **self-attention**
- http://nlp.seas.harvard.edu/annotated-transformer/

# Attention!

*The pink elephant tried to get into the car but it was too [?]*

-some words are indicative, such as *elephant*, *getting into the car*

-other words are less indicative, such as *pink*

- Paying attention to certain words in the sentence and ignoring others
  - Preceding words chime in with their opinions
  - But their contributions are weighted by how confident they are in their own expertise in predicting the next word

# Self-Attention Mechanism

- Attention: an information retrieval system
- Compute an attention score for each input token based on its relationship to all other tokens in the input sequence
- Three parameters: query, key, value
  - Query: a representation of the current task at hand
  - Key: representation of each word in the sentence, descriptions of the kinds of prediction tasks that each word can help with
  - Value: representation of the word in the sentence, unweighted contributions of each word

# How does it work?

- Each key is compared to the query using a dot product between each pair of vectors
  - The higher this number is for a particular key/query pair
  - The higher the resonance between query and key
  - The greater the contribution of value
- Resulting output is a sum of the values, weighted by the resonance between the query and each key

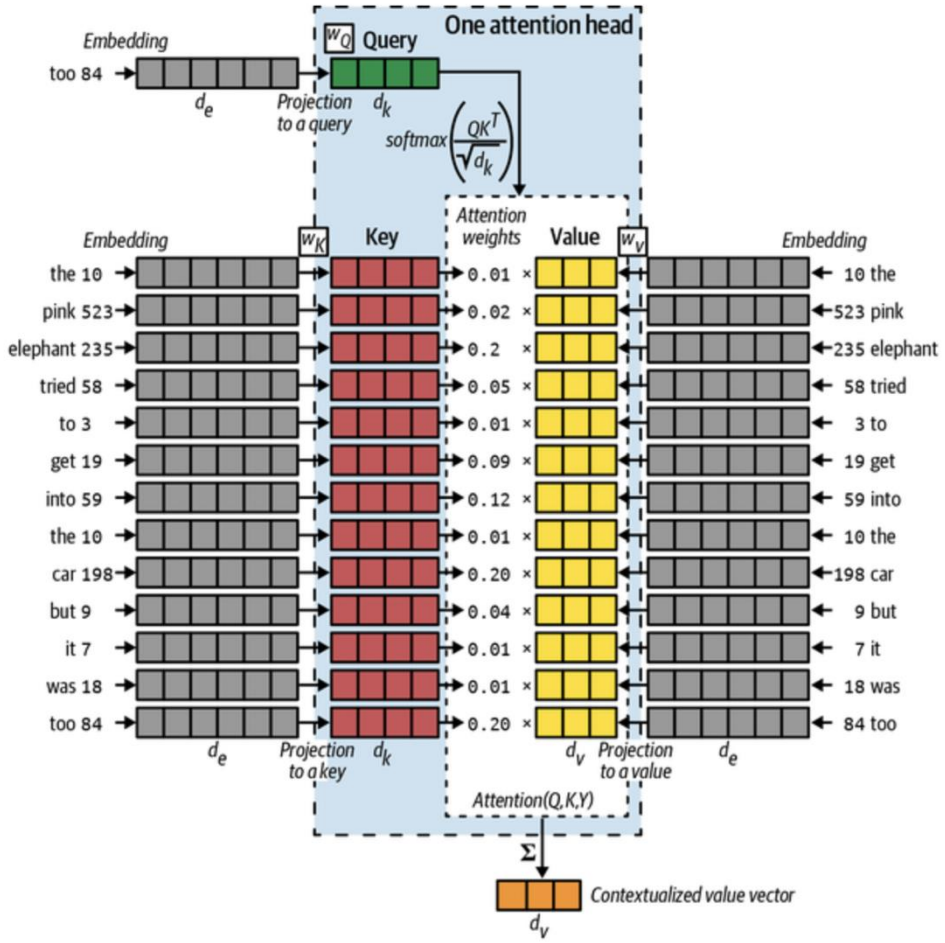


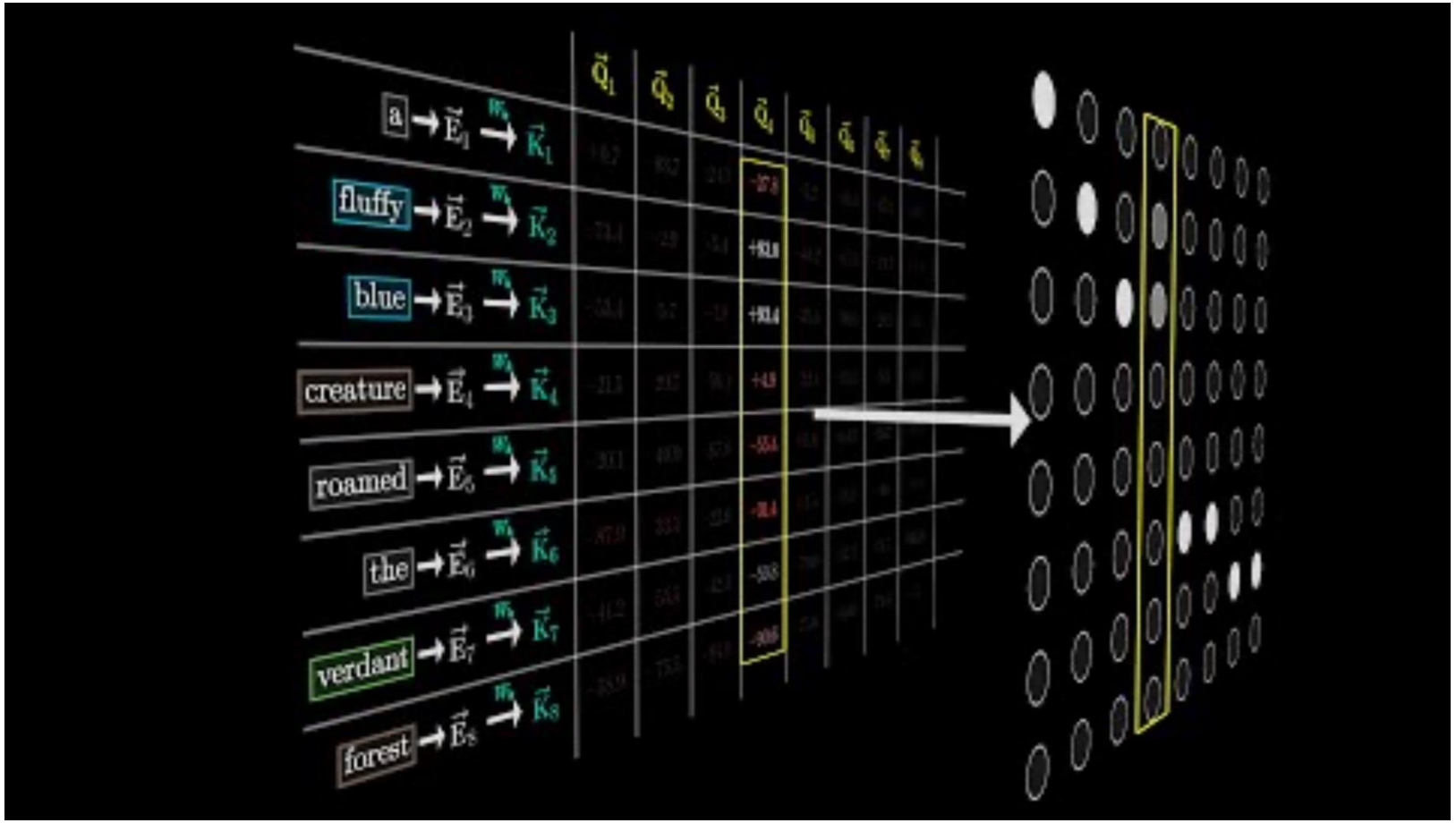

*Figure 9-2. The mechanics of an attention head*

# Animation Illustration

- Illustration

$$input\ embedding \times\ w_k = key$$
$$input\ embedding \times w_q = query$$
$$input\ embedding \times\ w_v = value$$
$$softmax\ (query\ \cdot\ key) = attention$$
$$sum\ (attention\ \times\ value) = output$$

# Video Illustration



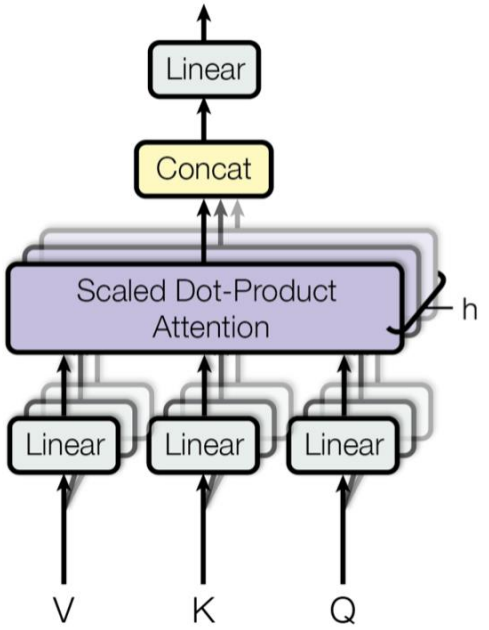

https://youtu.be/eMlx5fFNoYc?si=YaroMEnuwEsxIsHY

# Attention Offers Parallelism

- Before transformer, RNN (LSTM/GRU)is the state-of-art for sequence modeling
- RNN: challenging to parallelize, must process sequences one token at a time
  - Rely on hidden state to maintain all interesting information over the length of the sequence so that it can be drawn upon if required
  - A much less efficient approach
- Transformers: highly parallelizable, allowing them to be trained on massive datasets

# Multi-Head Attention

- Attend to different parts of input sequence in parallel
- Three steps:
  - Linear transformer: k-attention heads results in k query/key/value subspaces
  - Attention calculation: each attention head calculates attention scores
  - Attention aggregation: concatenate value vectors with weights (another set of parameters learned in training)
- Purpose: improve the efficiency
- Original paper: 8 parallel attention heads



# Transformer Block

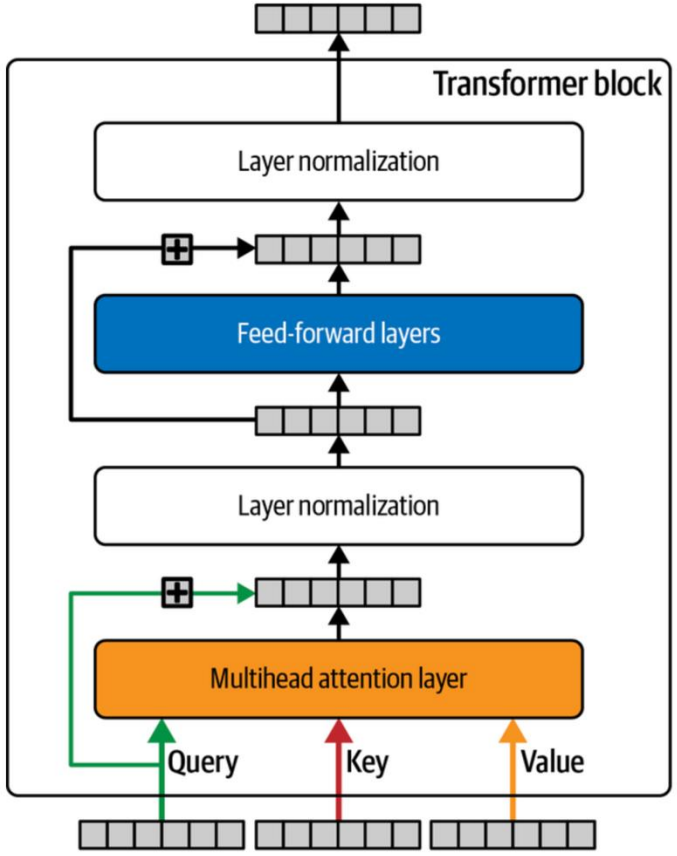



*Figure 9-6. A Transformer block*

# Positional Encoding

- Dot product between key and query is calculated in parallel not sequentially, but becomes a problem in for example:

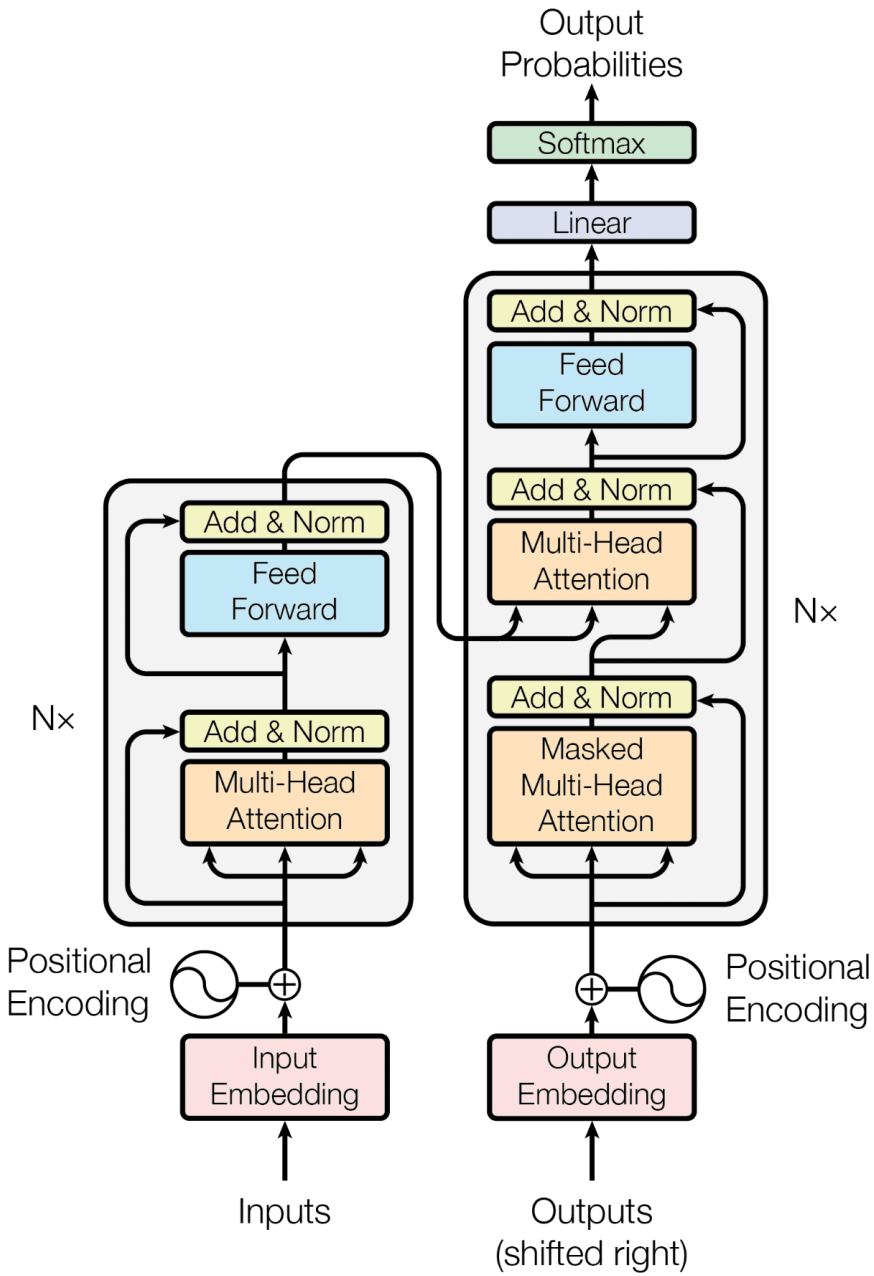
*The dog looked at the boy and [?]*

*The boy looked at the dog and [?]*

- Solution: positional encoding, token embedding + position embedding

# Transformer Architecture

- Encoder: a stack of 6 identical layers, each with two sub-layers
  - Multi-head self-attention mechanism
  - Position-wise fully connected feedforward net
- Decoder: a stack of 6 identical layers, each with three sub-layers
  - Multi-head attention over encoder output
  - Fully connected feedforward net
  - Self-attention modified with masking
    - Prevent positions to attending to subsequent positions, i.e., predictions for $i$ only depends on positions less than $i$



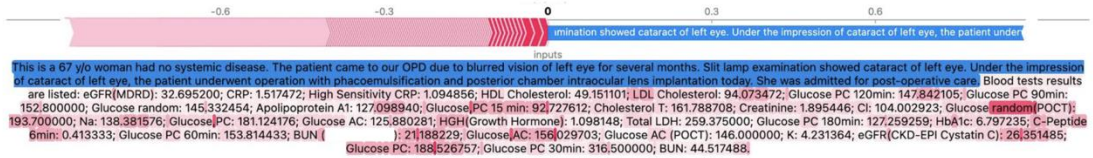
# Attention Offers Interpretability

- Deep learning models are often considered “black-box” models

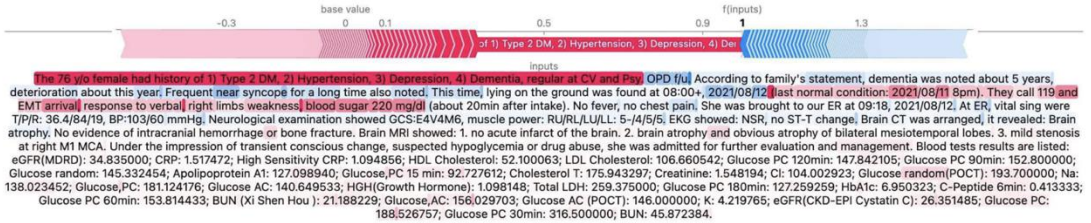
```
wine review : germany :
 pfalz:          51.53%
 mosel:          41.21%
 rheingau:       4.27%
 rheinhessen:    2.16%
 franken:        0.44%
 --------

wine review : germany : rheingau : riesling : this is a ripe , full - bodied
 riesling:       46.56%
 ,:      27.78%
 wine:           16.88%
 and:    4.58%
 yet:    1.33%
 --------

wine review : germany : rheingau : riesling : this is a ripe , full - bodied riesling
with a touch of residual sugar . it ' s a slightly
 sweet:          94.23%
 oily:           1.25%
 viscous:        1.09%
 bitter:         0.88%
 honeyed:        0.66%
 --------
```

*Figure 9-11. Distribution of word probabilities following various sequences*

**5.4 Interpretable Attention in Contextual Laboratory Test Values**




(a) Visualizing SHAP values in non-diabetes cases




(b) Visualizing SHAP values in diabetes cases

Figure 3: Comparing the interpretation sample of both cases diabetics and non-diabetic with Shapley Value

Large Language Multimodal Models

# Parameters Revisited

- Parameter: learned from training data

- Model size: number of parameters
  - “Size of brain”, capacity of learning
  - “Size of CPU”, capacity of computing

# Pre-training

- A core idea for large language models
- Process to teach models to recognize patterns and extract general features from large datasets
  - Basics of human language, grammar, sentence structure, basic facts about would
- Like K-12 students master the basics of reading, writing, mathematics, etc.
- But **unsupervised**

# Training Data

- Collection of examples relevant to the task that the model is being trained to perform
- "Transformer" paper: WMT 2014 English-German, 4.5 million sentence pairs; WMT 2014 English-French dataset, 36M sentences

- *Guess how many parameters?*

- *How long does it take to train?*

# Data Used in GPT-3 Training

| Dataset | Token | Weight in Training | Description |
|---|---|---|---|
| Common Crawl | 410 billion | 60% | Petabytes of data collected over 8 years of web crawling |
| WebText2 | 19 billion | 22% | Text of webpages from all outbound Reddit links |
| Books1 | 12 billion | 8% | Internet-based books corpora |
| Books2 | 55 billion | 8% | Internet-based books corpora |
| Wikipedia | 3 billion | 3% | Pages in English language |

https://www.linkedin.com/pulse/data-behind-chatgpt-khaled-abdelghani-pmp-cdmp/

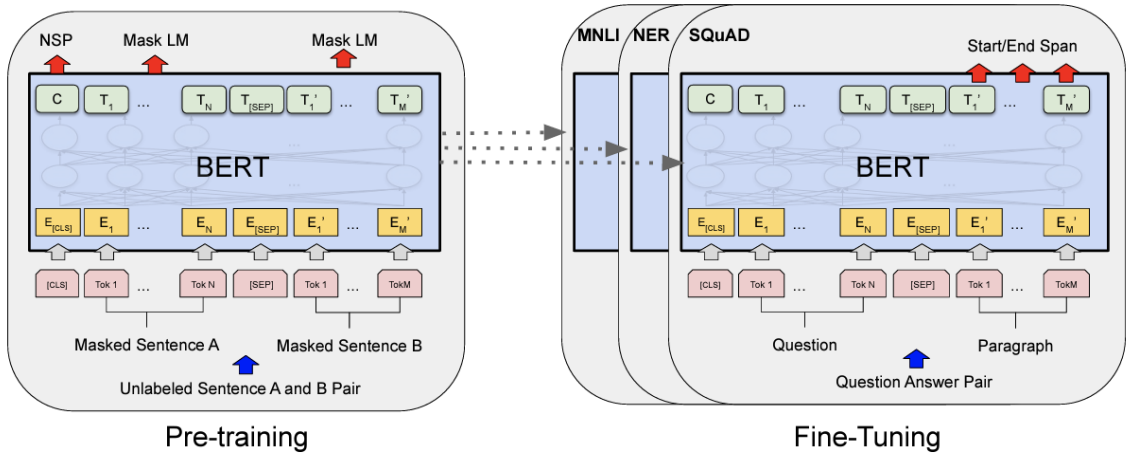
# Pre-training Approaches

Table 9-1. The three Transformer architectures

| Type | Examples | Use cases |
|---|---|---|
| Encoder | BERT (Google) | Sentence classification, named entity recognition, extractive question answering |
| Encoder-decoder | T5 (Google) | Summarization, translation, question answering |
| Decoder | GPT-3 (OpenAI) | Text generation |

# BERT

- Bidirectional Encoder Representation from Transformer
  - Developed by Google in 2018
- Pre-trained with two tasks
  - Masked language modeling: randomly masking some words in input, predict the masked words
  - Next sentence prediction: predict whether a pair of sentences are consecutive
- Task-specific output layer can be added for specific task



# BERT Use Cases

- Sentiment analysis
  - https://huggingface.co/models?pipeline_tag=text-classification&sort=downloads&search=sentiment
- Finance
  - ESG BERT: sustainable investing
- Healthcare
  - ClinicalBERT: trained on EHR notes
  - BEHRT ("BERT for EHR"): represent a patient's medical history as a sequence, where each visit or entry is like a "sentence" of coded data
  - Med-BERT: pretrained contextualized embeddings on structured EHR data
  - BioBERT: trained on PubMed articles

# GPT and ChatGPT

- Radford, Alec, et al. "Improving language understanding by generative pre-training." (2018)
- GPT-1: 2018
  - 117 million parameters, 1GB training data
- GPT-2: 2019
  - 1.5 billion parameters, 40GB training data
- GPT-3: 2020
  - 175 billion parameters, 580GB training data
- ChatGPT: 2022

# Fine-Tuning

- After pre-training, focus on more specific data, for specific tasks
  - Update parameters to specialize to tasks, or new domains
- Like specializing in a particular field or profession in college education after K-12

- Two critical developments: instruction tuning and RLHF

# Fine-Tuning (Cont.)

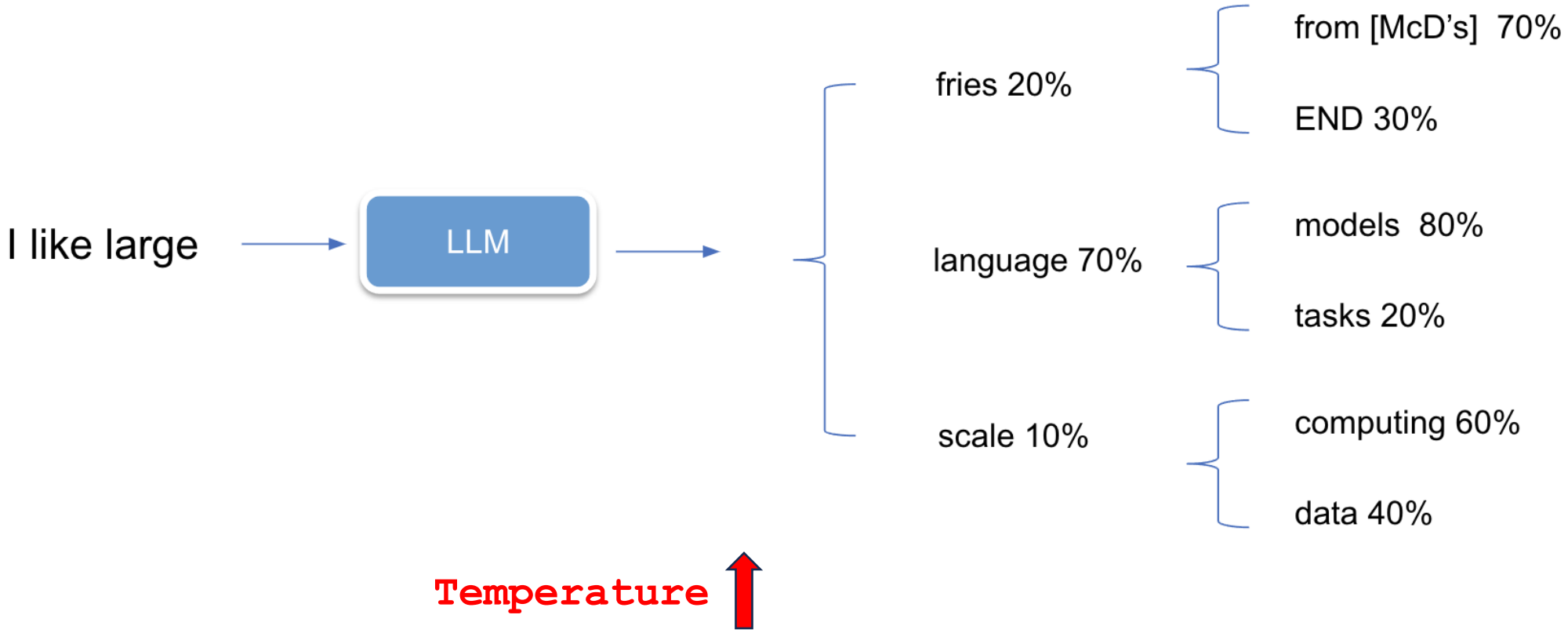
- Instruction tuning
  - Fine-tuning language models on a collection of tasks described via instructions
  - **Supervised**: input data is labeled

- **Reinforcement** Learning from Human Feedback (RLHF)
  - Align an intelligent agent to human preferences
  - human annotators rank LLM-generated outputs from the same prompt
- OpenAI hired contractors to annotate data during training of GPT-3

# Temperature

- Controls the randomness of predictions
- Low Temperature: produce more predictable text
  - Output becomes deterministic and repetitive, as it tends to choose the most likely next word each time
- High Temperature: increase the chances of less likely words
  - Make the text more diverse and creative, but also more random and potentially nonsensical
- *Why "temperature"*
  - In thermodynamics, temperature controls the distribution of energy states of particles in a system

# “I like large” Example

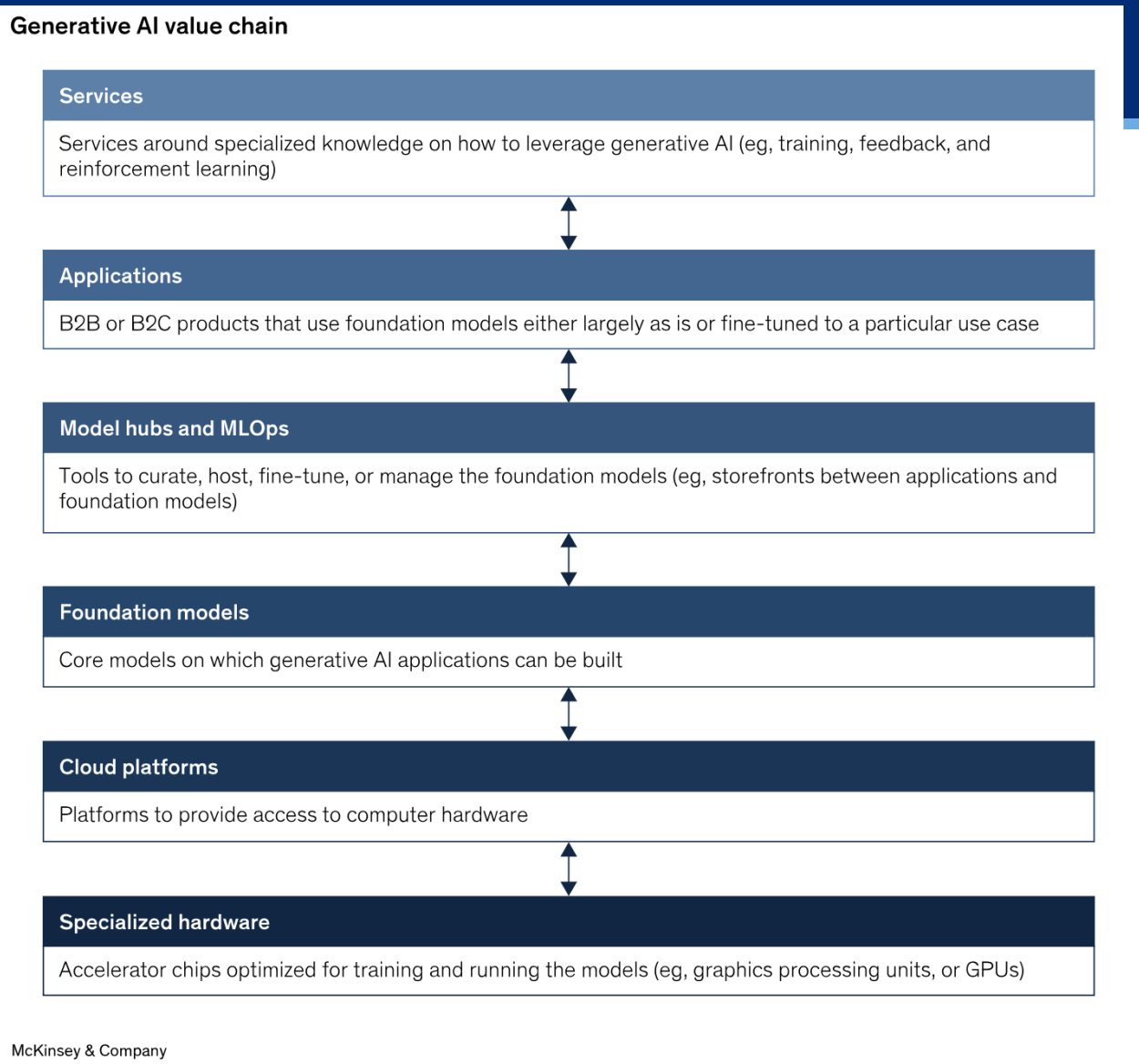


# Lab 1 Attention Mechanism

- Build a mini-gpt using wine review data
- Focus will be on attention and how sequence is generated

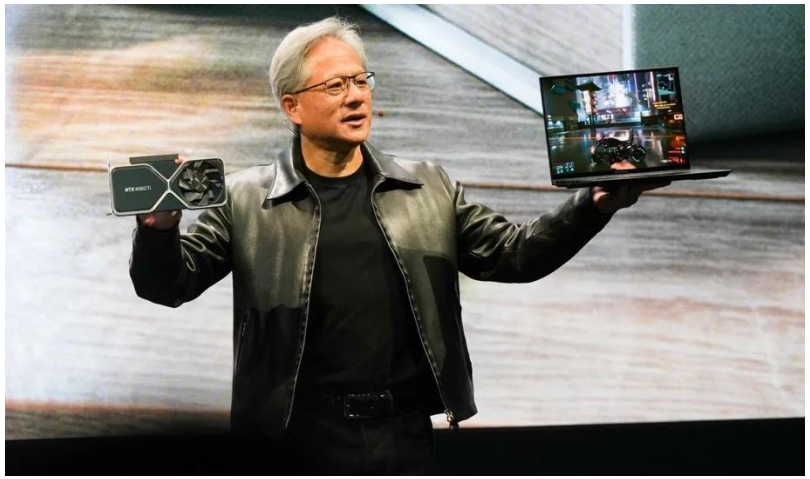
JOHNS HOPKINS
CAREY BUSINESS SCHOOL

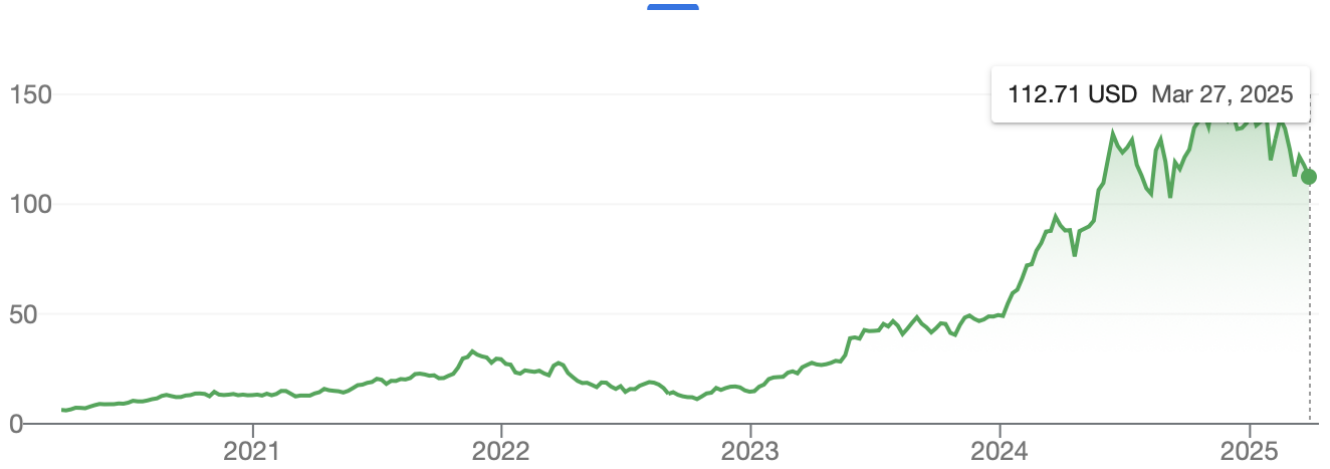
# Generative AI Ecosystem


Generative AI value chain
Services
Services around specialized knowledge on how to leverage generative AI (eg, training, feedback, and reinforcement learning)
Applications
B2B or B2C products that use foundation models either largely as is or fine-tuned to a particular use case
Model hubs and MLOps
Tools to curate, host, fine-tune, or manage the foundation models (eg, storefronts between applications and foundation models)
Foundation models
Core models on which generative AI applications can be built
Cloud platforms
Platforms to provide access to computer hardware
Specialized hardware
Accelerator chips optimized for training and running the models (eg, graphics processing units, or GPUs)
McKinsey & Company

# Hardware: GPUs

- Graphics Processing Units (GPUs)
  - Specialized chips with highly parallel structures
  - Process a large number of similar calculations simultaneously
- *Best-known GPU producer?*



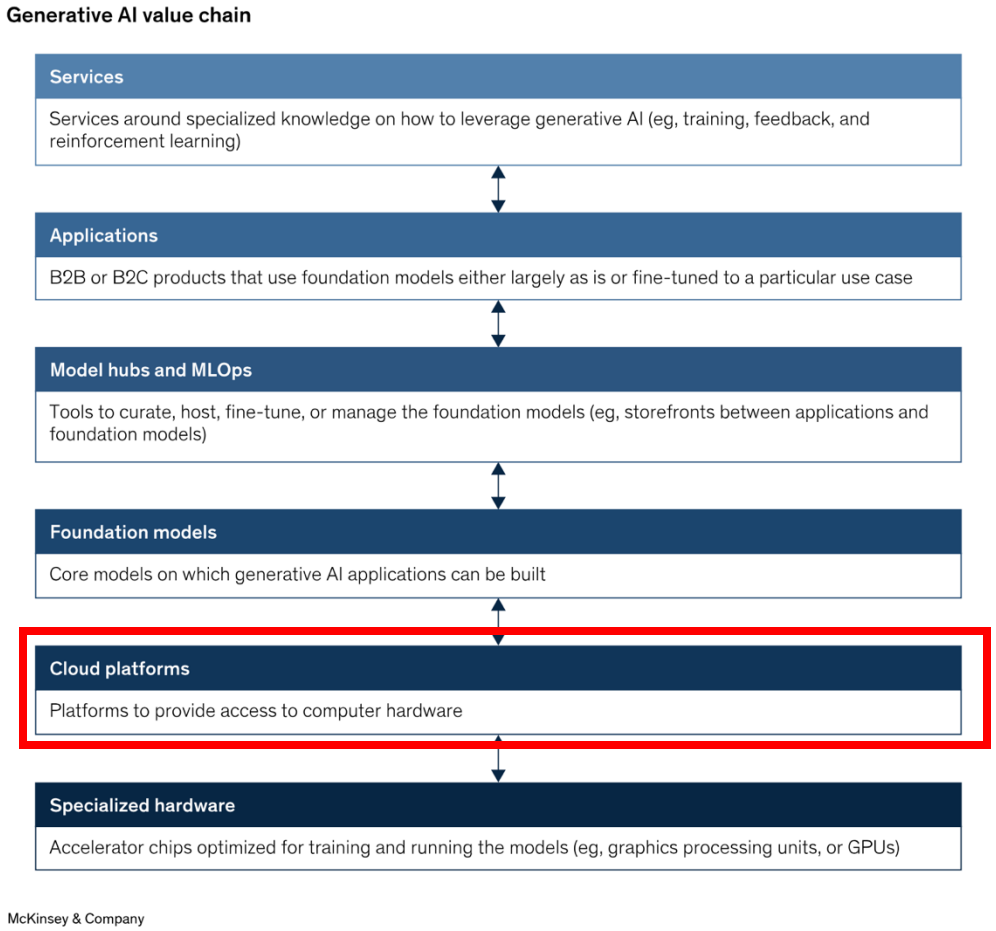


# GPUs (Cont.)

- Performance measure: Floating Point Operations Per Second (FLOPS)
  - Number of discrete arithmetic calculations in one second
- Computational power grow exponentially
  - 2020 A100: 19.5 FP32 TeraFLOPS
  - 2022 H100: 67 FP32 TeraFLOPS
  - …
- Cost factors: macroeconomic conditions, demand, trends, re-seller opportunism, performance improvements in GPU models

# Cloud Platforms

- Powered by high-performance GPUs
- Commonly available through major vendors, e.g., AWS, Google Cloud and Microsoft Azure
- In-house approach can be cost-prohibitive and impractical
  - Purchase GPUs and related hardware to set up the infrastructure to train and run locally

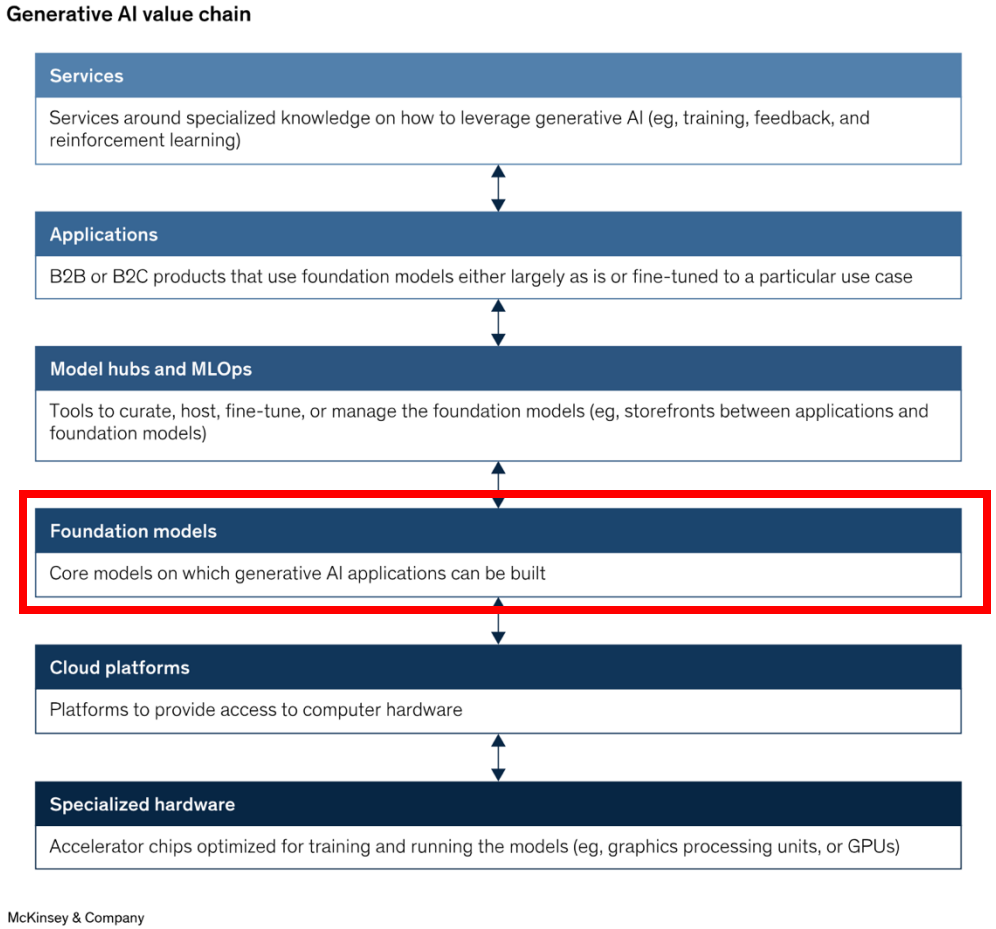


# Energy Concerns

- Significant demand for natural resources
  - In Jan 2023, 13 million daily unique visits to ChatGPT, estimated electricity costs were $50,000 a day
  - Electricity cost of training GPT-3 alone could reach $12 million
  - Google: training 540-billion-parameter PaLM model cost 3.4 gigawatt-hours over about 2 weeks
    - Equivalent to powering about 300 US households for a year
- Compute and energy consumption of deep learning models for inference had improved over time
- Small language models for cost-effectiveness

# Foundation Models

- Transformers trained on massive data sets without being optimized for specific domain or downstream tasks
- Will introduce major foundation models in the next class
- Data, computing resources, technical expertise required to create and train high-performing models **form** a significant barrier to entry



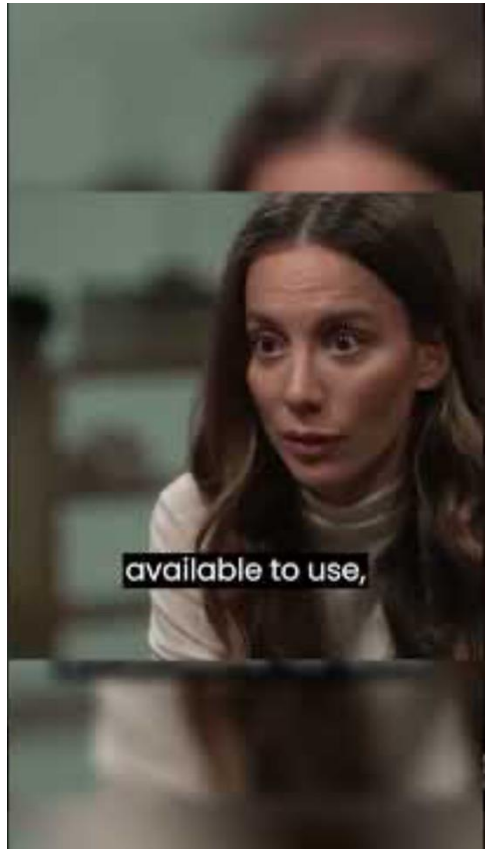
# *Lessons from the Cloud?*

- MIT Sloan study (2024): market of foundation models could consolidate among a few players
- Three reasons:
  - Cost and capability required to create, sustain, and improve infrastructure
  - Demand-side network effects
    - ChatGPT's large user base attracts developers to build plug-ins
    - Data from user input create feedback loop to improve
  - Economies of scale
    - Negotiate better rates from GPU vendors and cloud service providers
    - Lower cost for users
- *Need to be proofed, why?*

# Data Copyright Law

- Training data: public or proprietary
- Training corpora for generative AI systems may contain copyrighted works

- Lawsuits: https://www.bakerlaw.com/services/artificial-intelligence-ai/case-tracker-artificial-intelligence-copyrights-and-class-actions/



# Guardrails and Content Policy

- Final step in the deployment of a generative-AI system
- Add additional instructions or restrictions
  - Before sending queries to model
  - Or after generating a response
  - *Which one is better?*
- Microsoft Tay case
  - https://www.youtube.com/watch?v=qMe8bXp8RfA

# Applications

- Built on top of foundation or fine-tuned models to serve a specific use case

- GitHub Copilot: software development
- Sudowrite: creative writing
- Evisort: draft legal contracts

# Applications (Cont.)

- Jumpcut Media: summarize books and movie screenplays
- Alltius: technical troubleshooting, customer support
- CarMax: use Microsoft Azure OpenAI service
  - Read and synthesize over 100,000 customer reviews for each vehicle make, model and year
  - Generate 5,000 easy to read summaries
  - Would have taken editorial team 11 years to complete

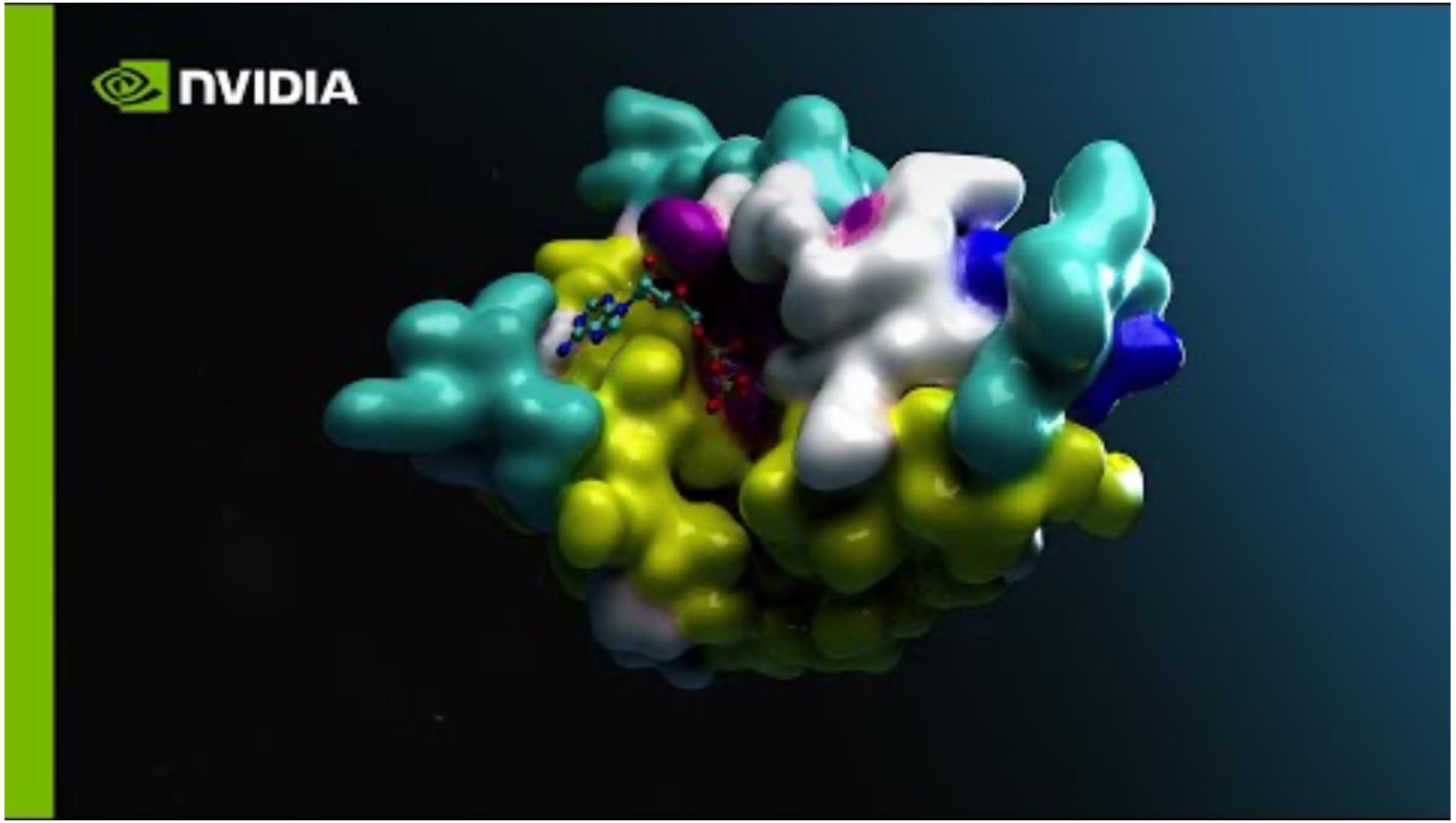
# Service Matters

- Functionality of applications can be easily replicated by competitors
- Need to distinguish yourselves
- Model vs Product, service matters
  - User experience
  - Interface
  - Customer service
  - Market segmentation
  - …
- Large and loyal user base has advantage

# Domain-Specific Opportunities

- Organizations having access to large volumes of high-quality data in specific domains have the advantage
  - BloombergGPT in class 4
- Other sections, insurance, media, and health care, are likely to benefit as well

# Case: Drug Discovery



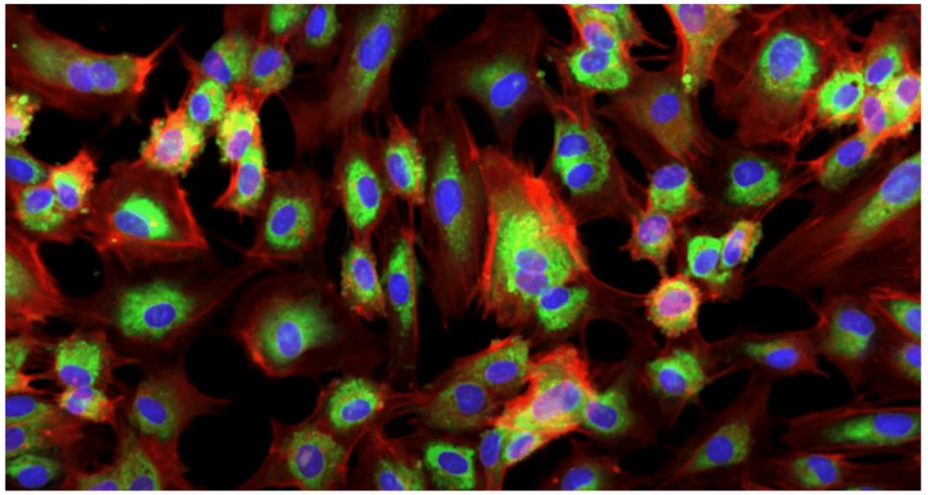
# NVIDIA BioNeMo

- Curate more precise field of drug candidates to investigate
- Reduce need for expensive and time-consuming physical experiments

**NVIDIA Generative AI Is Opening the Next Era of Drug Discovery and Design**

NVIDIA BioNeMo, which is fueling the computer-aided drug discovery ecosystem, now features more than a dozen generative AI models and cloud services.

January 8, 2024 by Kimberly Powell



https://blogs.nvidia.com/blog/drug-discovery-bionemo-generative-ai/

# BioNeMo Ecosystem

- A generative AI platform providing services to develop, customize and deploy foundation models for drug discovery
- Ecosystem of partners
  - Biopharmaceutical company Recursion
    - Offer one foundation model to BioNeMo
  - Biotech company Terray Therapeutics
    - Use BioNeMo for AI model development
  - Protein engineering and molecular design companies Innophore and Insilico Medicine
    - Bring BioNeMo into computational drug discovery applications
  - Biotech software company OneAngstrom and systems integrator Deloitte
    - Use BioNeMo cloud APIs to build AI solutions for clients

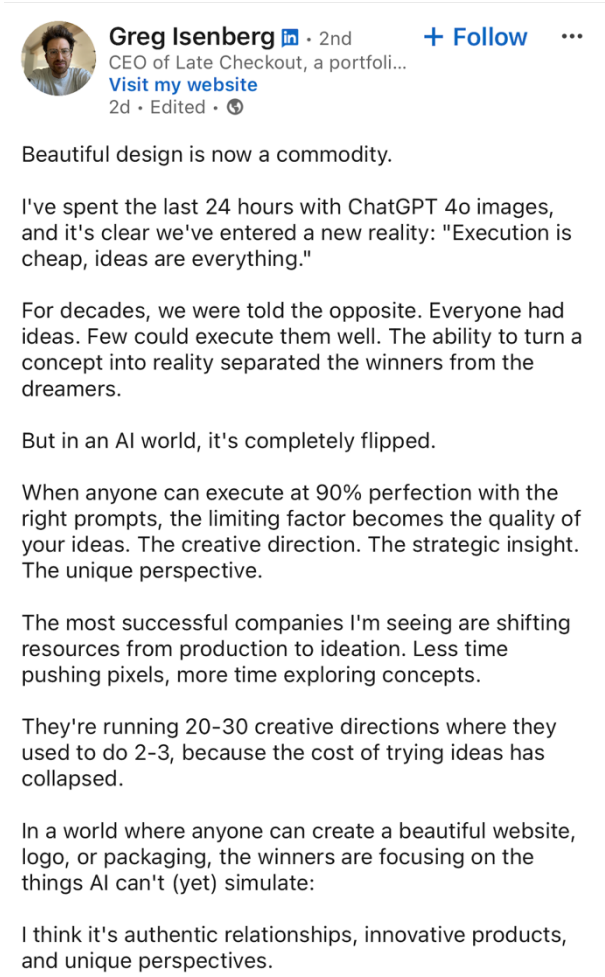
# How does it work

- Idea: generate or design novel molecules likely to process desired properties
- Pretrained biomolecular AI models for
  - Protein structure prediction
  - Protein sequence generation
  - Molecular optimization
  - Generative chemistry
  - Docking prediction
  - etc.

# Foundation Models Included

- Models invented by NVIDIA
  - MolMIM generative chemistry model for small molecule generation
- Open-source models by research teams, curated by NVIDIA
  - OpenFold protein prediction AI
- Proprietary models developed by partners
  - Recursion's Phenom-Beta for embedding cellular microscopy images

# Preview



Greg Isenberg · 2nd + Follow ···
CEO of Late Checkout, a portfoli...
Visit my website
2d · Edited ·

Beautiful design is now a commodity.

I've spent the last 24 hours with ChatGPT 4o images, and it's clear we've entered a new reality: "Execution is cheap, ideas are everything."

For decades, we were told the opposite. Everyone had ideas. Few could execute them well. The ability to turn a concept into reality separated the winners from the dreamers.

But in an AI world, it's completely flipped.

When anyone can execute at 90% perfection with the right prompts, the limiting factor becomes the quality of your ideas. The creative direction. The strategic insight. The unique perspective.

The most successful companies I'm seeing are shifting resources from production to ideation. Less time pushing pixels, more time exploring concepts.

They're running 20-30 creative directions where they used to do 2-3, because the cost of trying ideas has collapsed.

In a world where anyone can create a beautiful website, logo, or packaging, the winners are focusing on the things AI can't (yet) simulate:

I think it's authentic relationships, innovative products, and unique perspectives.

- *"Execution is cheap, ideas are everything."*
- *When anyone can execute at 90% perfection with the right prompts, the limiting factor becomes the quality of your ideas. The creative direction. The strategic insight. The unique perspective.*
- *But at the high end, there's a premium on the truly unexpected - the ideas an AI wouldn't generate because they break conventional patterns.*
- *The challenge for most of us now isn't "how do we execute this idea?" but "which ideas are actually worth executing?"*

# Next Week

- Large Language Models and Strategies
- Reasoning Models
- Prompt Engineering

# References


- Andy Wu and Matt Higgins. "Generative AI Value Chain." Harvard Business School Background Note 724-355, July 2023. (Revised July 2023.)
- Kartik Hosanagar and Ramayya Krishnan. "Who Profits the Most From Generative AI?." MIT Sloan Management Review 65.3 (2024): 24-29.