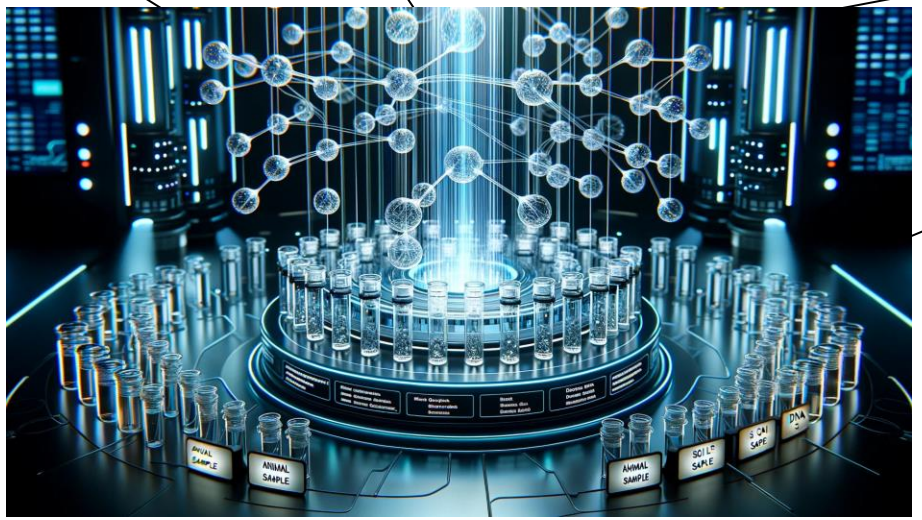
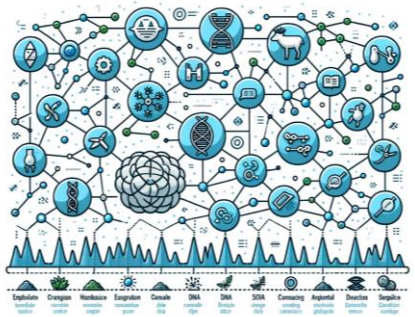


https://github.com/Phillips-Lab-MTSU/DL-Workshop

EXPLORING DEEP LEARNING FOR SCIENTIFIC DISCOVERY

Joshua L. Phillips

Associate Professor
Department of Computer Science
Program in Data Science
Program in Computational and Data Science
Middle Tennessee State University



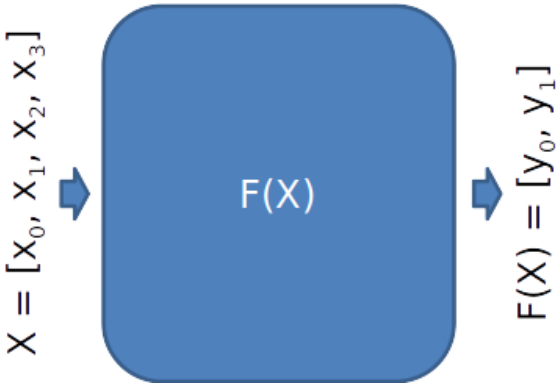
ChatGPT 4 (DALL-E 3) Prompt: "Generate an image of a neural network processing multiple short strands of DNA from microbes taken from animal and soil samples." (2023-10-18)

MACHINE LEARNING BASICS

Q: How do we write computer programs that can solve problems which are *very* difficult?
A: We don't. We write computer programs that *learn* to solve problems for which it is very difficult to write computer programs.

Learning Paradigms

- Unsupervised Learning
- Supervised Learning
- Reinforcement Learning
- Adversarial Learning



ML algorithms need three things for learning:

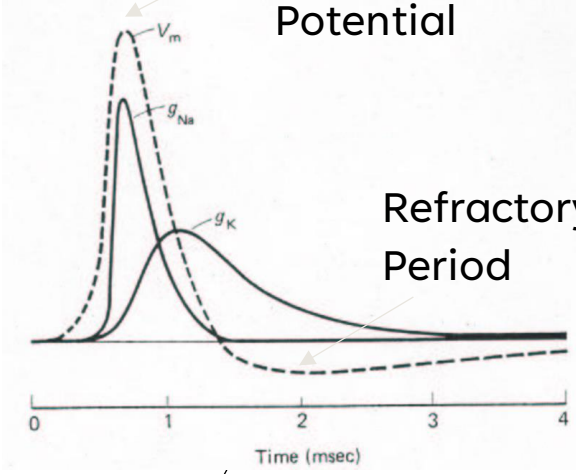
1. Task
2. Performance Measure
3. Experience

Deep Learning uses an architecture built of artificial neurons – if the neurons are wired together correctly, task is solved

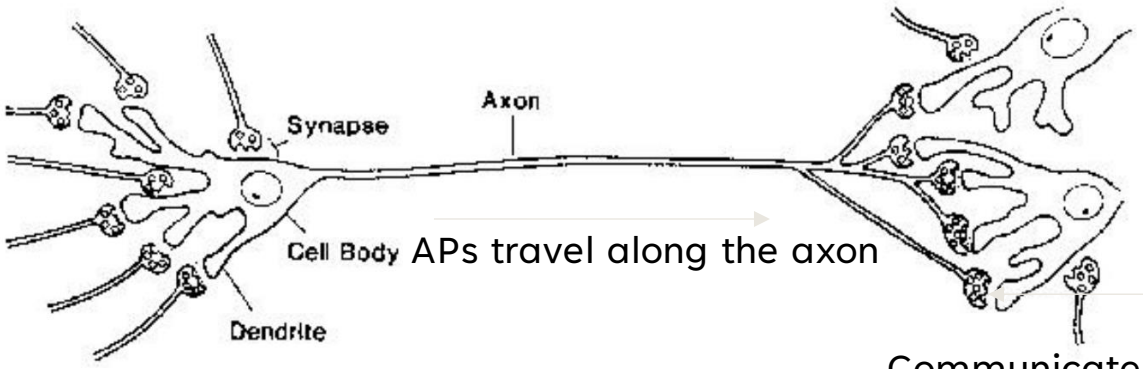
Biological Inspirations

- Neurons communicate:
  - *electrically* via **action potentials**
  - *chemically* via **neurotransmitters**

Action Potential



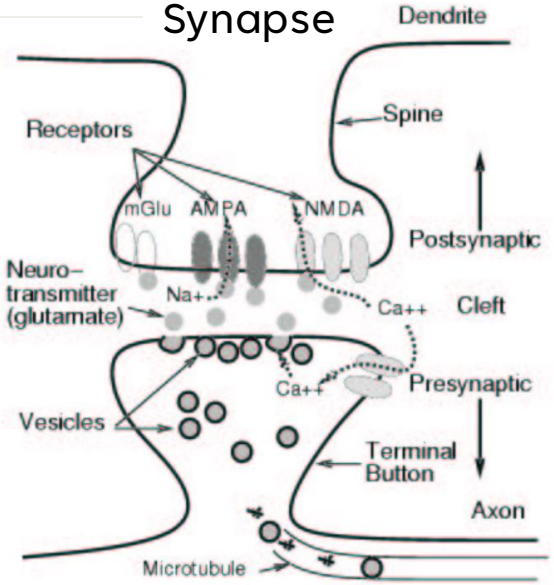
Refractory Period



APs travel along the axon

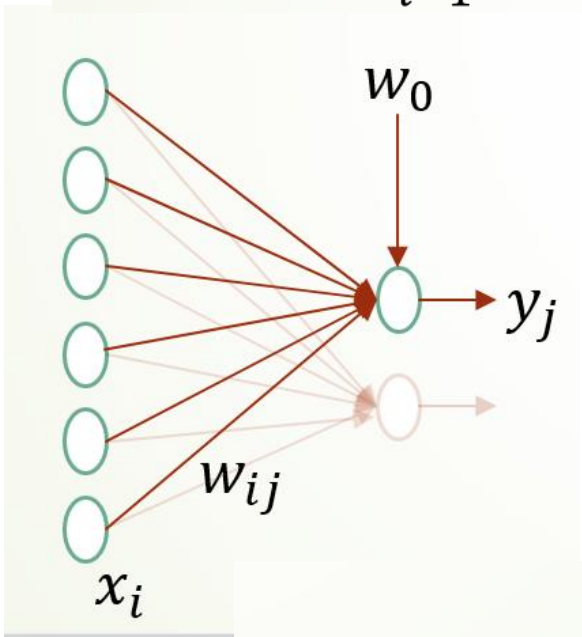
Communicate chemically at the synapse

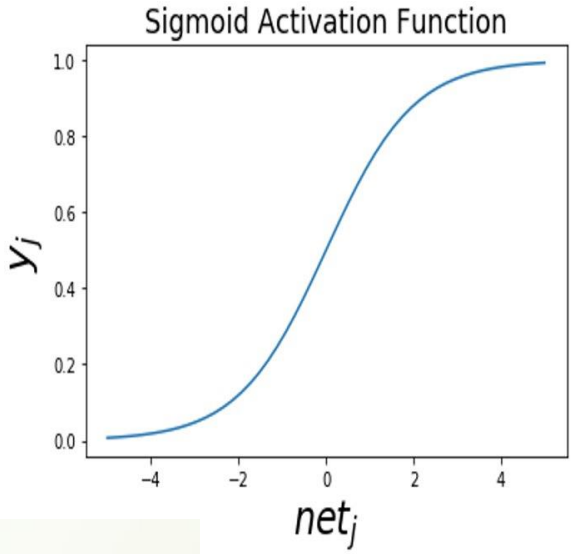
Synapse



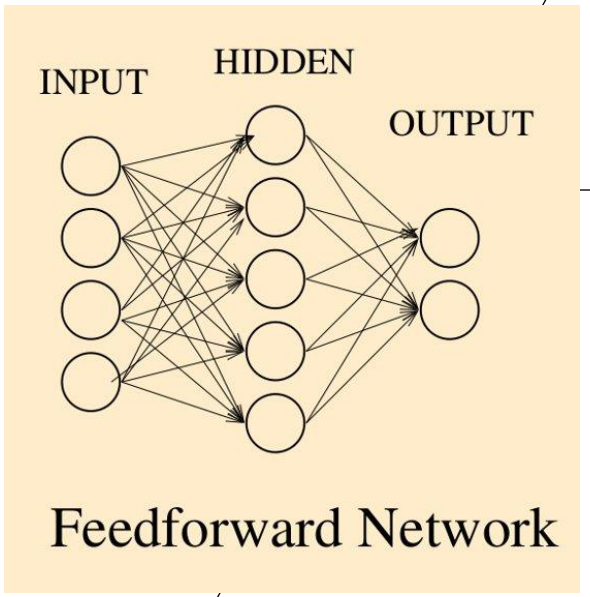
Artificial Neural Networks

$$net_j = w_0 + \sum_{i=1}^{n} x_i w_{ij}$$

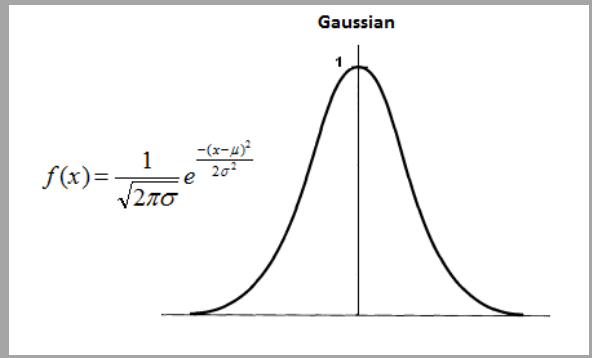




$$y_j = \frac{1}{1 + e^{-net_j}}$$



Something for later:
Radial Basis
Activation Functions



Error

Learning can be viewed as a search for connection weights which optimize the error function…

$$E(\boldsymbol{W}) = \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{c}(y_k(\boldsymbol{x}^n;\boldsymbol{w}) - t_k^n)^2$$

We want to search for the weights which produce the low error for the training data…

Weights

Error

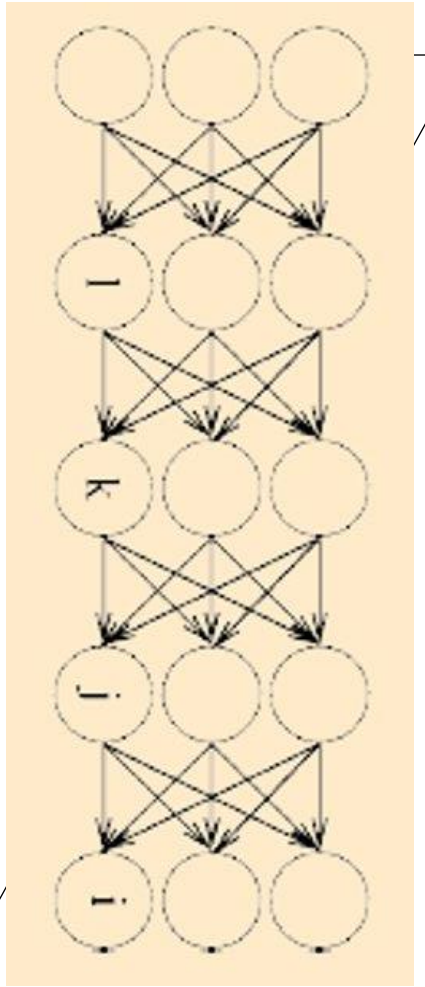
The timestep, t, is our current epoch… $\Delta \boldsymbol{W}^t \propto -\nabla_{\boldsymbol{w}} E(\boldsymbol{W}^t)$

$$\boldsymbol{W}^{t+1} = \boldsymbol{W}^t + \Delta \boldsymbol{W}^t$$

$$w_{kj}^{t+1} = w_{kj}^t - \alpha \left(\frac{\partial E}{\partial w_{kj}}\right)_{\boldsymbol{W}^t}$$

Following ∇ would **maximize** the error but following -1* ∇ would **minimize** the error (what we want!)

Weights

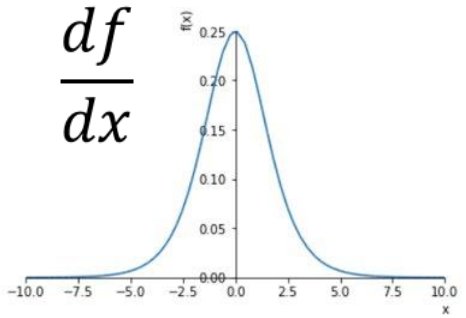


$$\delta_j = g'(net_j)\sum_i w_{ij}\delta_i$$

$$\delta_k = g'(net_k)\sum_j w_{jk}\delta_j$$

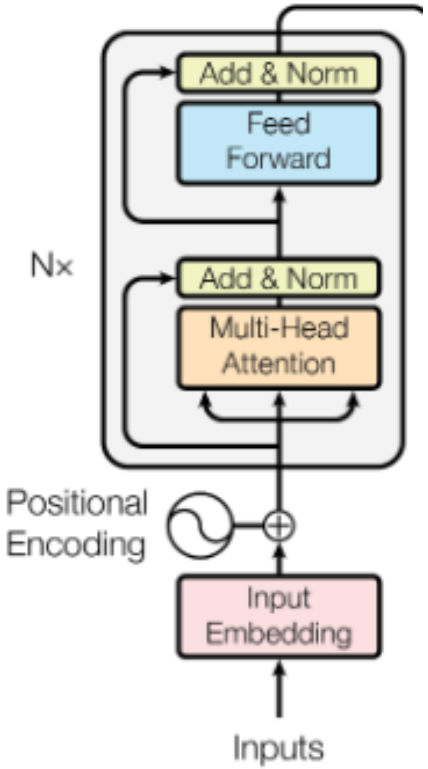
$$\delta_l = g'(net_l)\sum_k w_{kl}\delta_k$$

$$\frac{df}{dx}$$

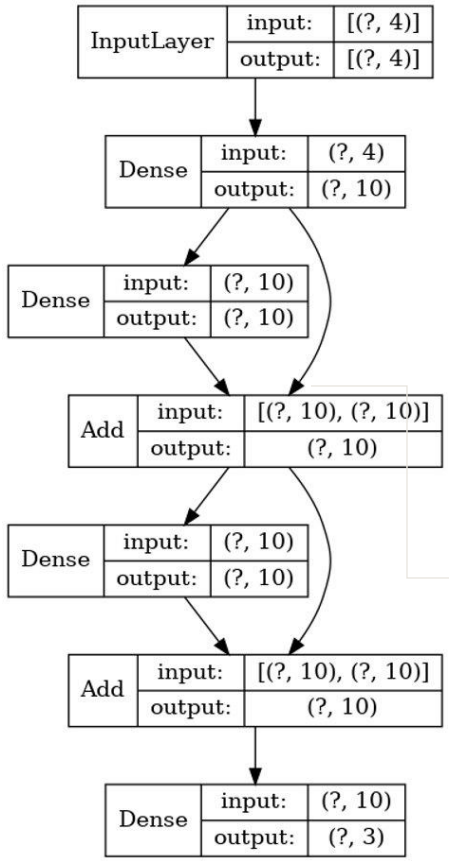


Deep Learning

Vaswani et al., 2017



Transformer Architecture

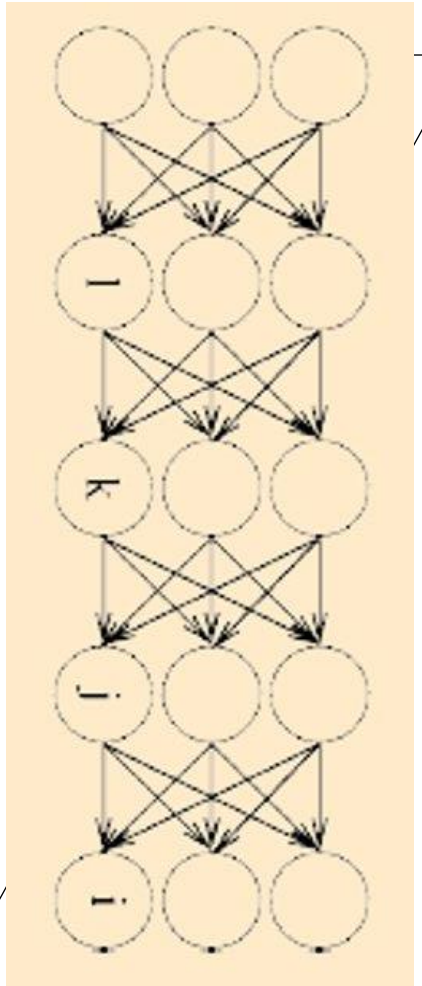


Nonlinear Residual

Nonlinear Residual

Fixed

He et al. 2015

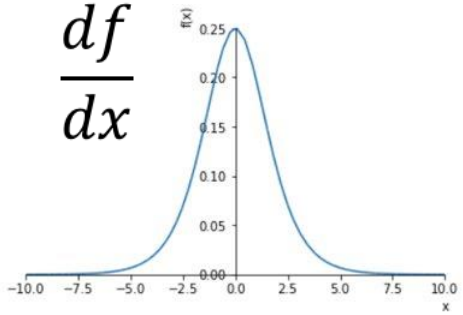


$$\delta_j = g'(net_j)\sum_i w_{ij}\delta_i$$

$$\delta_k = g'(net_k)\sum_j w_{jk}\delta_j$$

$$\delta_l = g'(net_l)\sum_k w_{kl}\delta_k$$

$$\frac{df}{dx}$$

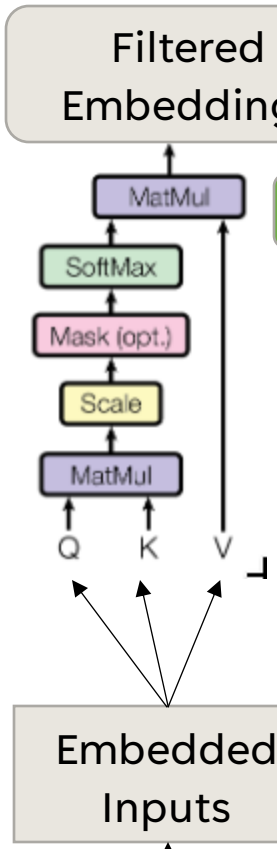


Number of layers is not as important as ***deep residual structure*** to say that the model is a form of ***deep learning***

TRANSFORMER: WHY ATTENTION HELPS

Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).

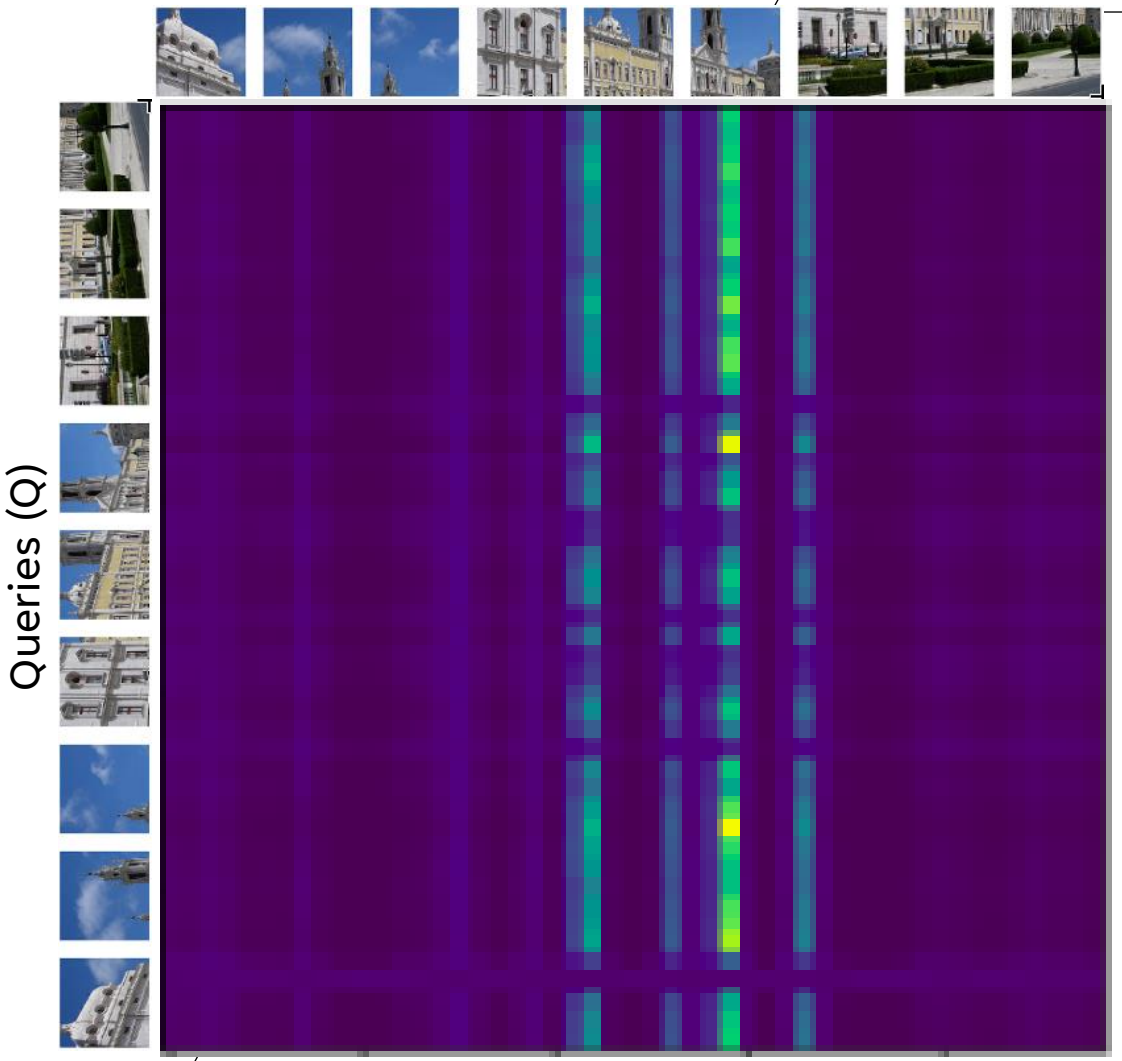
TRANSFORMER: SELF-ATTENTION MECHANISM



Embedding

1. Embeddings can be linear, CCNs, etc.
2. Split into Queries, Keys, Values
3. All Queries are compared to All Keys
4. Matches allow Values to pass forward
5. Non-matching Values are filtered out

(All choices on what to **self-attend** to using Q and K are learned based on **feedback**)



Vaswani, Ashish, et al. "Attention is all you need." arXiv preprint arXiv:1706.03762 (2017).

BERT PRETRAINING METHODOLOGY

Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

NLP Examples:

BERT, mBERT, RoBERTa

___ ___ cat ___ ____

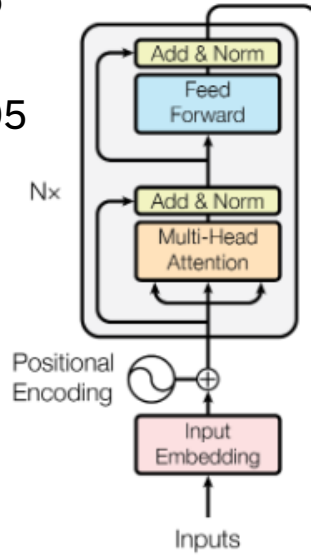
___ ___ ____ _____ __ _ cat

Transformer(s)

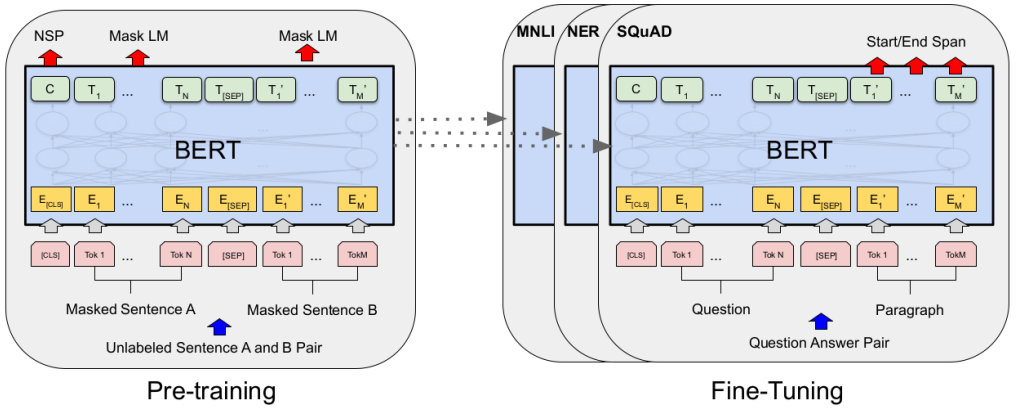
[CLS] The ___ ran fast

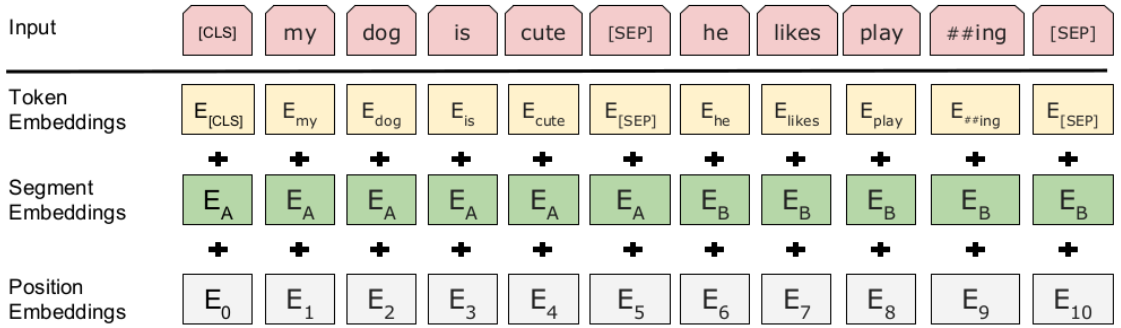
[CLS] The game ended in a ___

Better *contextualized meaning* from these models



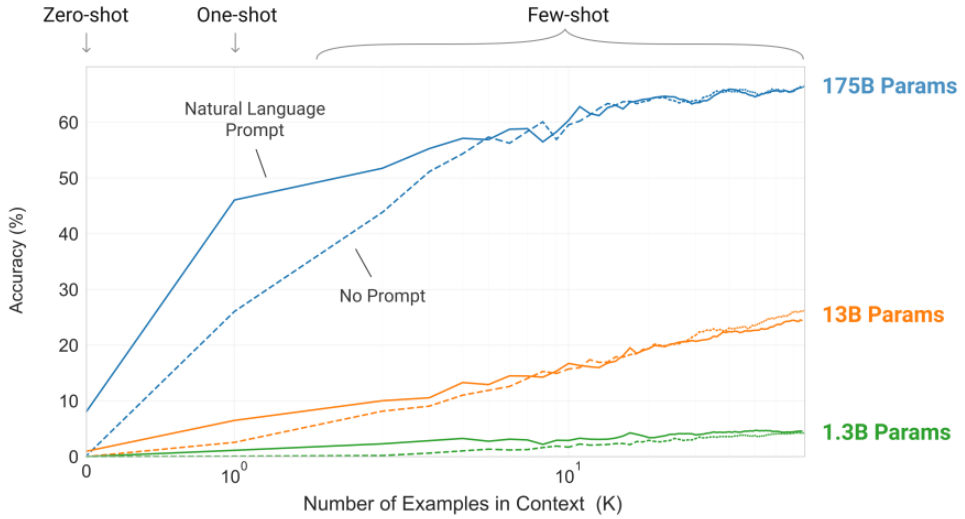
Pretrained embedding model





GPT (GENERATIVE PRETRAINED TRANSFORMER)

- Vaswani et al., 2017 - **Transformer** architecture
- Radford et al., 2018 and Brown et al., 2020
- Simple *generative training* and *testing* procedure, perfectly suited for the *transformer* architecture.
- Very large model, very large data set

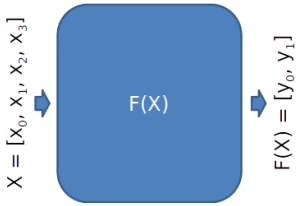


GPT-3 (Brown et al. 2020)

[To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:]
One day when I was playing tag with my little sister, she got really excited and she started doing these crazy farduddles.

[A "yalubalu" is a type of vegetable that looks like a big pumpkin.
An example of a sentence that uses the word yalubalu is:]
I was on a trip to Africa and I tried this yalubalu vegetable that was grown in a garden there. It was delicious.

1. The
2. Cat
3. Ran
4. Fast



1. Cat
2. Ran
3. Fast
4. <STOP>

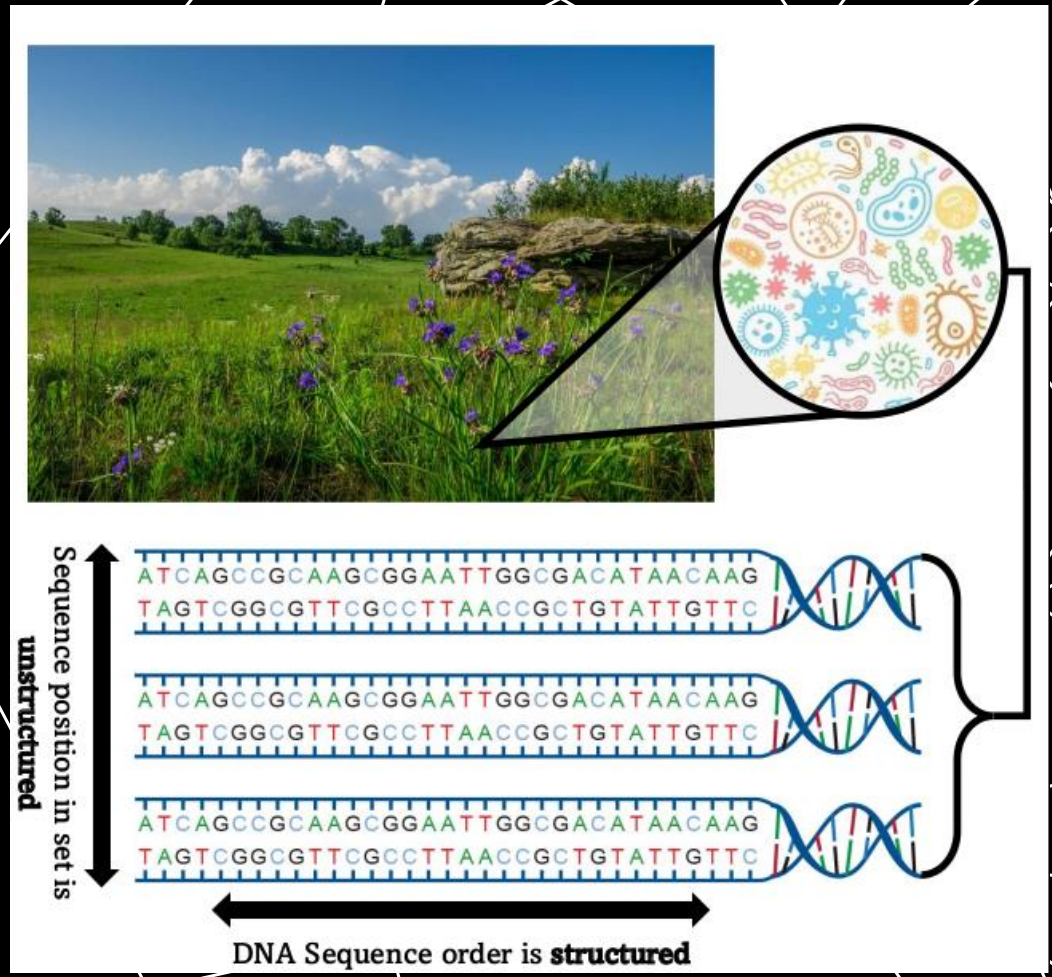
The [P(duck), P(cat), P(fast), P(no), ...]
The cat [P(duck), P(cat), P(ran), ...]
The cat ran [P(fast), P(quickly), P(slowly, P(no) ...]

SETBERT: DEEP LEARNING FOR OMICS DOMAINS

- We began studying high-throughput sequencing data (HTS) in 2020
- NSF Award# **1933925** (**Collaborative Research: RoL: Impacts of plants and communities on soil microbial composition and function across phylogenetic scales)**
- Structured – DNA sequences are *ordered* sequences
- Unstructured – HTS FASTQ are *unordered* sets of sequences

David Ludwig
COMS Ph.D. Student

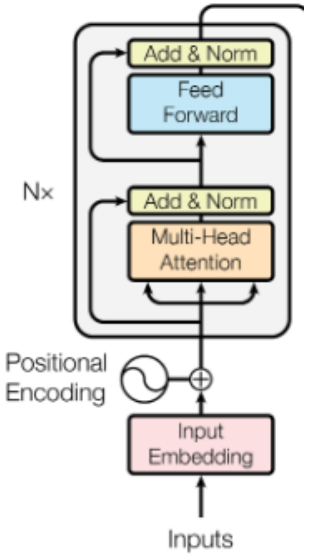
```
TACGTAGGGTGCAAGC
AACGTAGGTACCGAGC
TACGAAGGGGGCTAGC
TACAGAGGGTGCAAGC
TACGTAGGTGGCAAGC
GACGTAGGGTGCGAGC
        …
TACGGAGGGGGCTAGC
GACGAACCGTCCGAAC
CACGGACCGCACGAAC
```



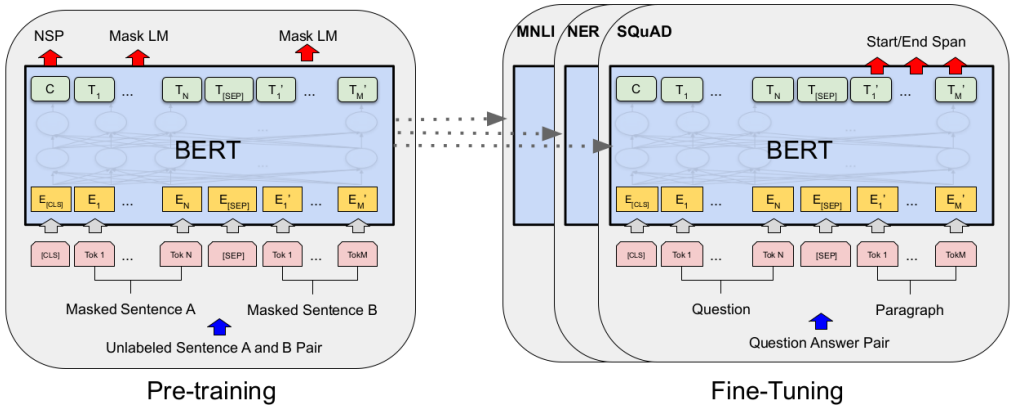
BERT PRETRAINING METHODOLOGY

Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

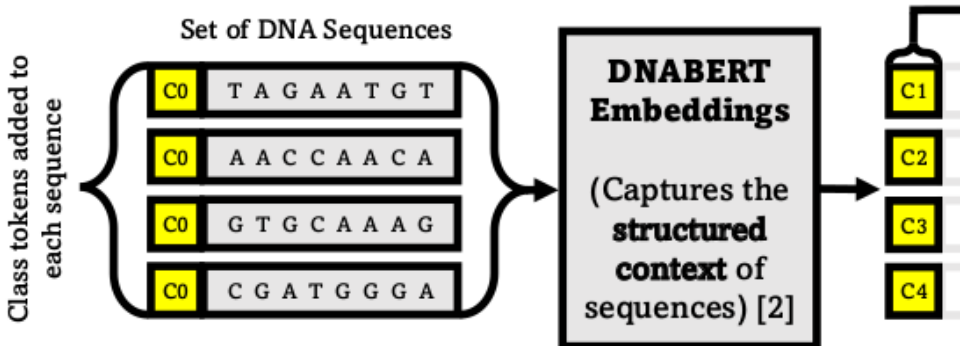
NLP Examples:
BERT, mBERT, RoBERTa

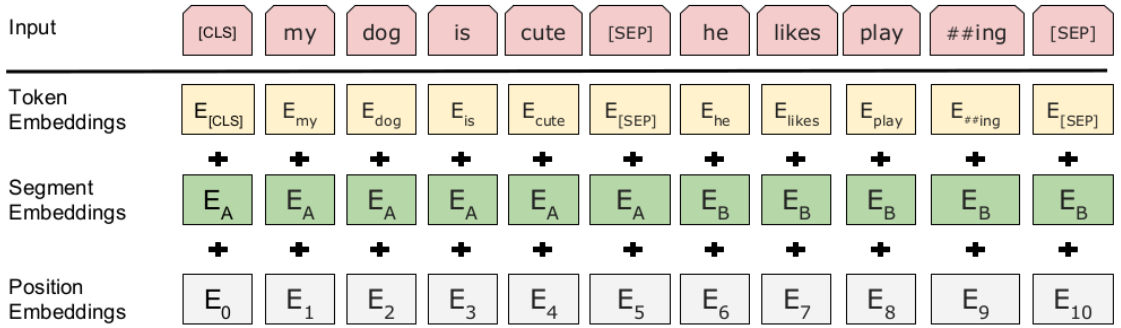


Pretrained embedding model



Analogously applied to HTS data...

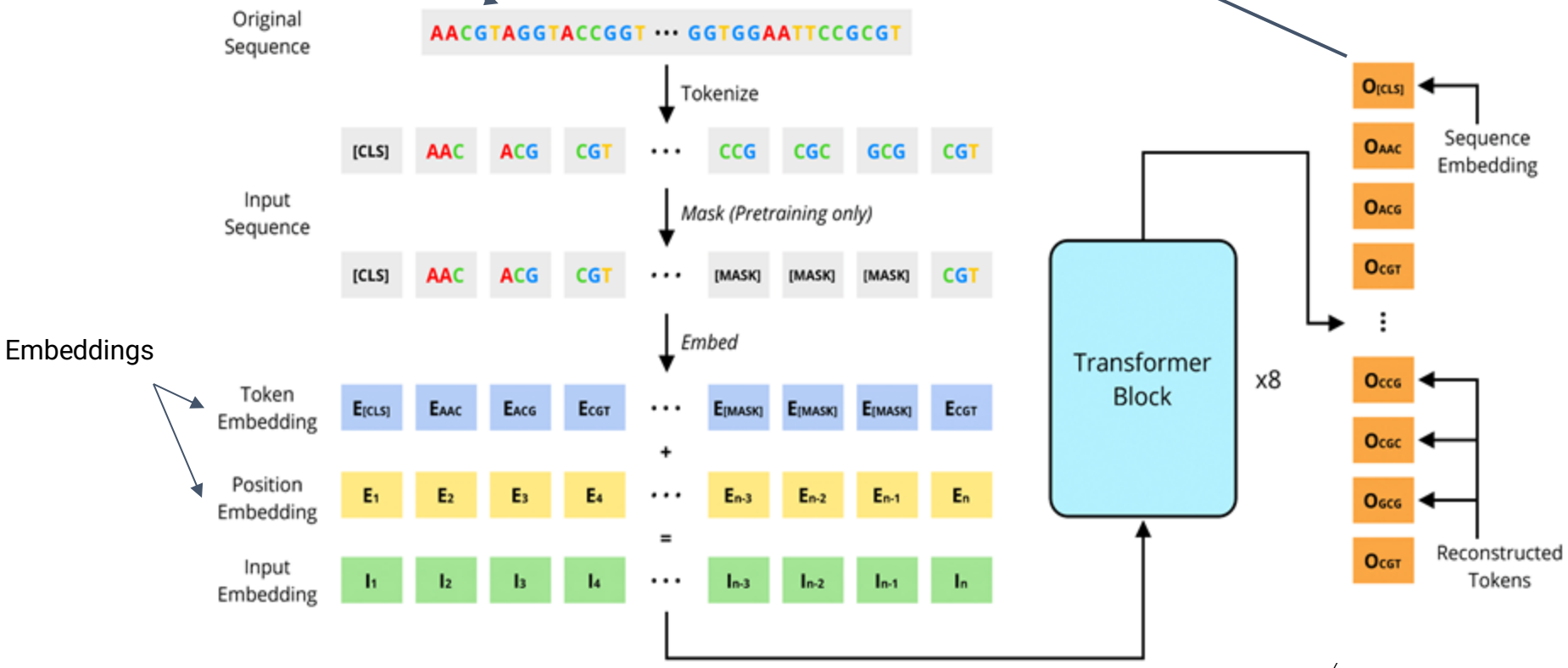




DNABERT: SINGLE SEQUENCE FOUNDATIONS

Map raw DNA sequence to a high-dimensional vector embedding

AACGTAGGTACCGGT•••GGTGGAATTCCGCGT ⟶ <0.1, -0.3, 0.2, 0.2, •••, 0.4, 0.9, -0.2>



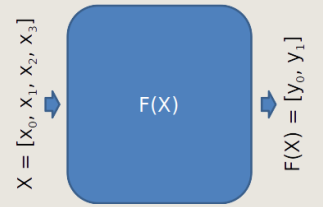
Ji et al., *Bioinformatics*, Volume 37, Issue 15, August 2021, Pages 2112–2120, https://doi.org/10.1093/bioinformatics/btab083

BERT Training Methology

- Vaswani et al., 2017 - **Transformer** architecture
- Devlin et al., 2018

The ___ ran fast

The game ended in a ___



___ cat ___ _____

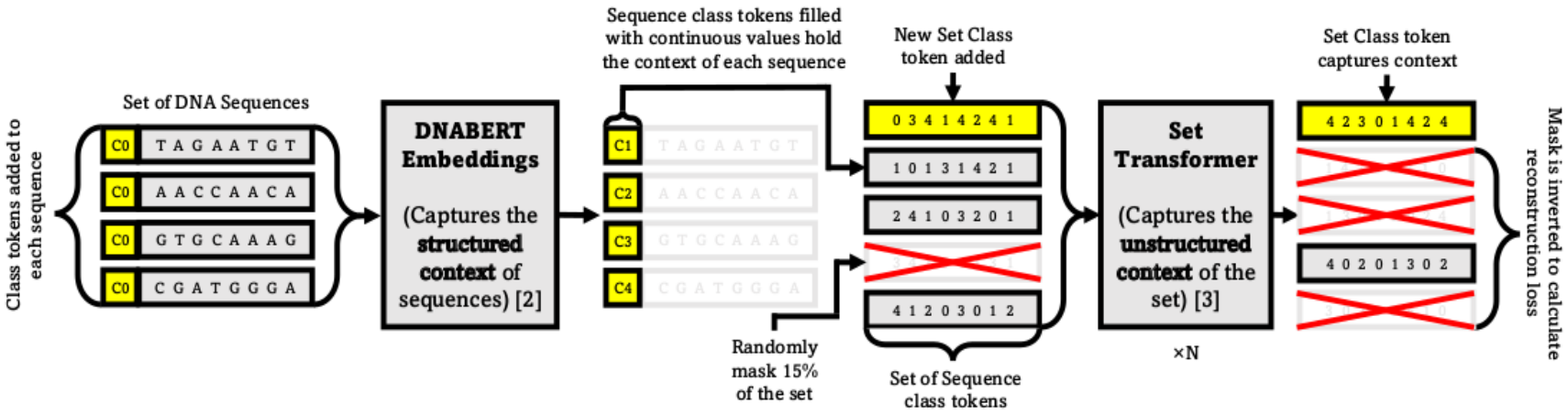
___ _____ _____ __ _

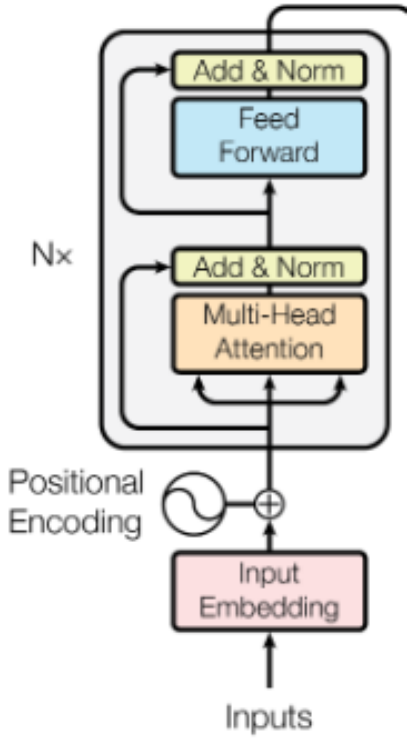
cat

Better *contextualized meaning* from these models

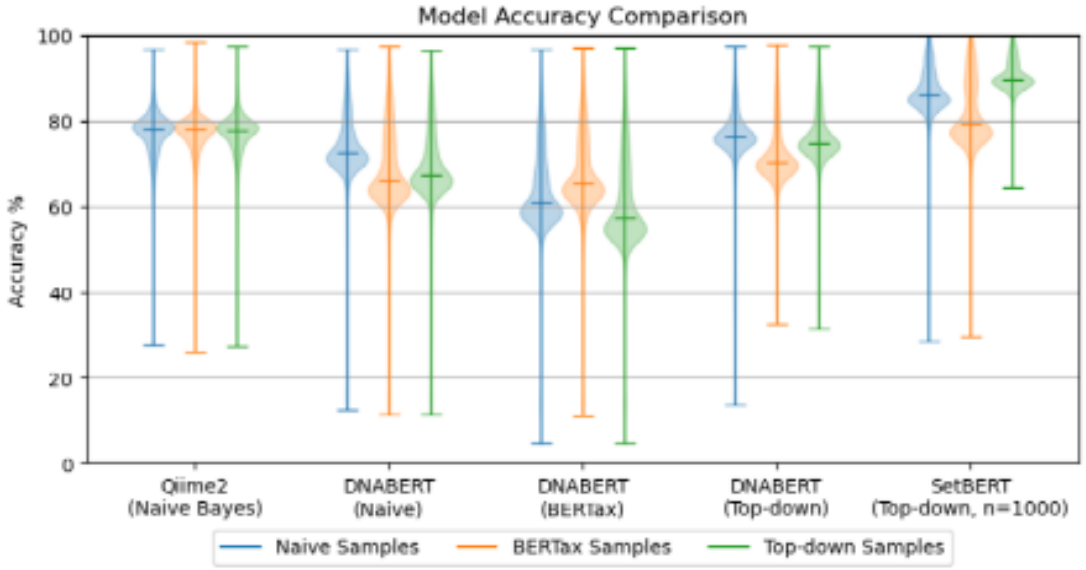
SETBERT: DNABERT + SET TRANSFORMER (LEE ET AL., 2019)

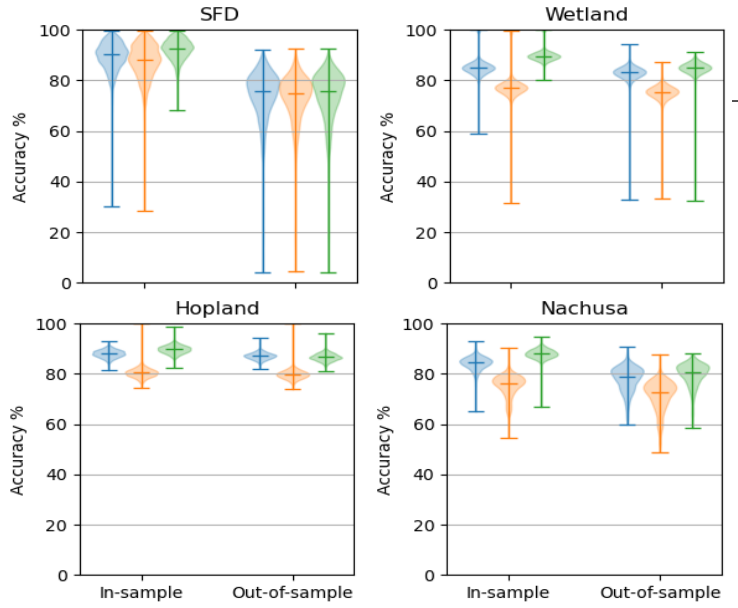
- The Set Transformer component integrates single-sequence embeddings in a *permutation-equivariant* manner
- Removal of *position encodings* is all that is needed to make the MHA module of Transformers perform in this manner
- We construct an HTS sample as a set of embeddings provided by our pretrained DNABERT module
- Pretraining can be performed for sets of sequences using fixed DNABERT embeddings
- Fine-tuning can then be performed for any downstream task
- Changes in attention between pretrained embeddings (or compared to fine-tuned results) can be made, and attention attribution is also compatible

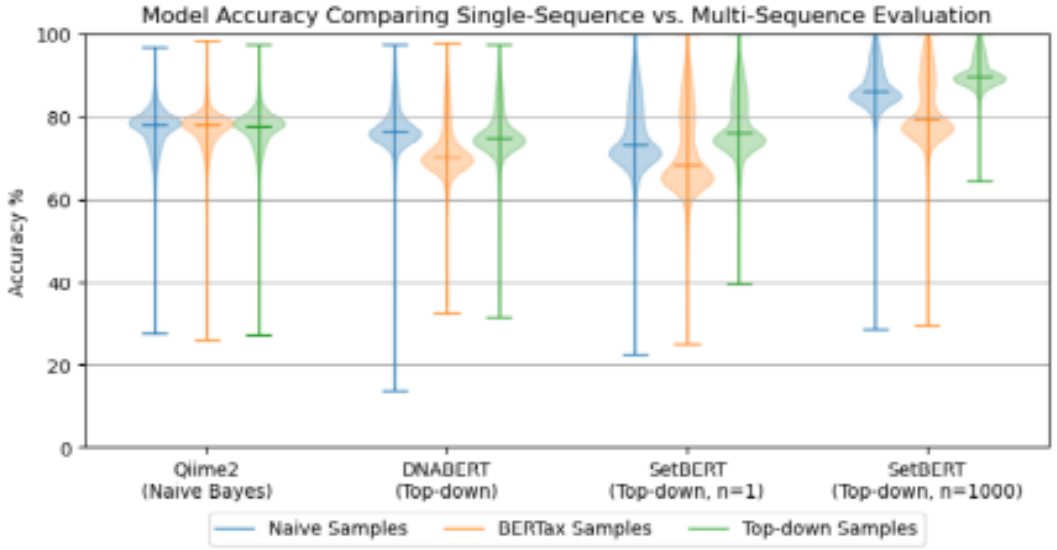


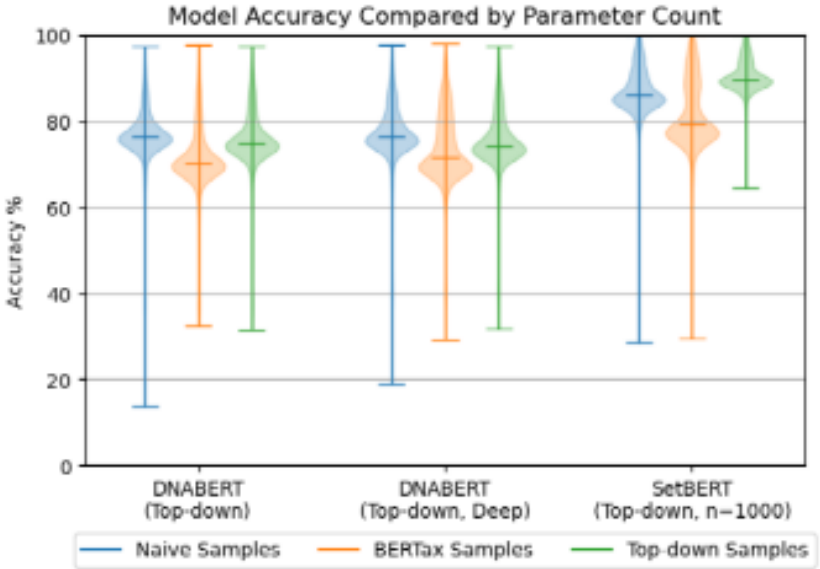


TAXONOMIC PREDICTION: GENUS LEVEL

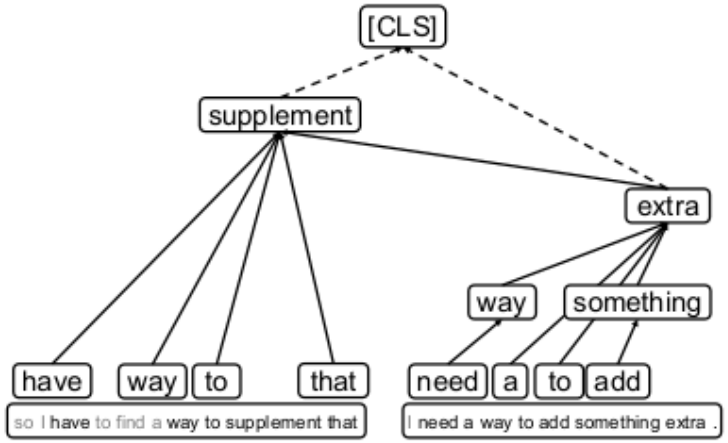




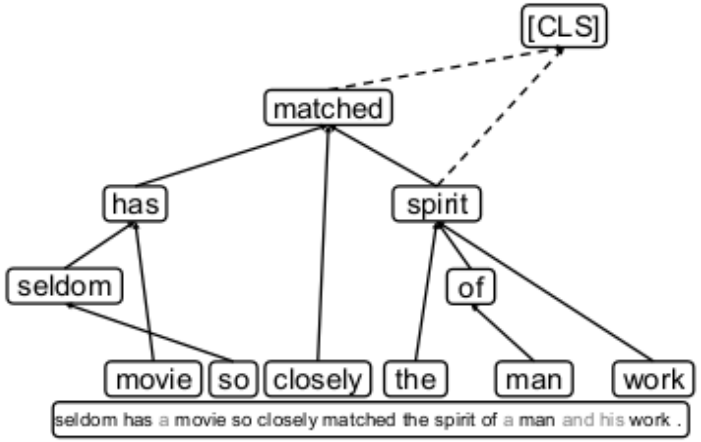




INTERPRETING ATTENTION MECHANISMS



(a) Example from MNLI



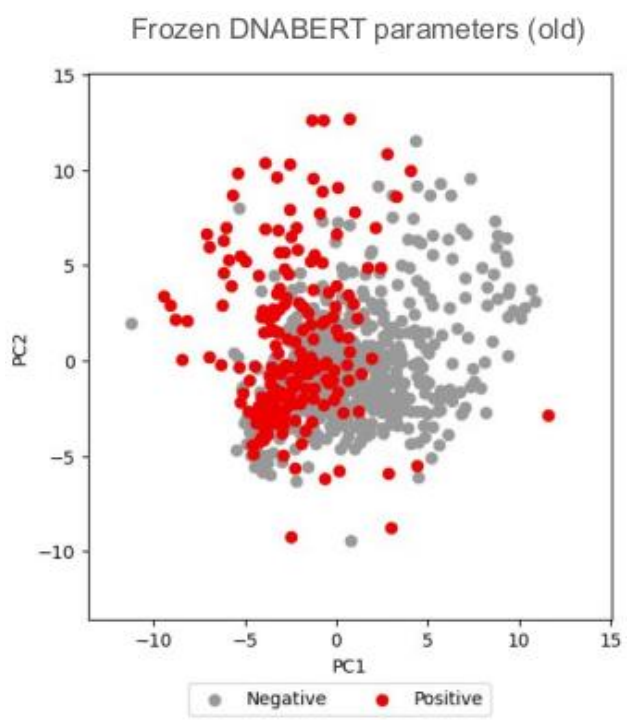
(b) Example from SST-2

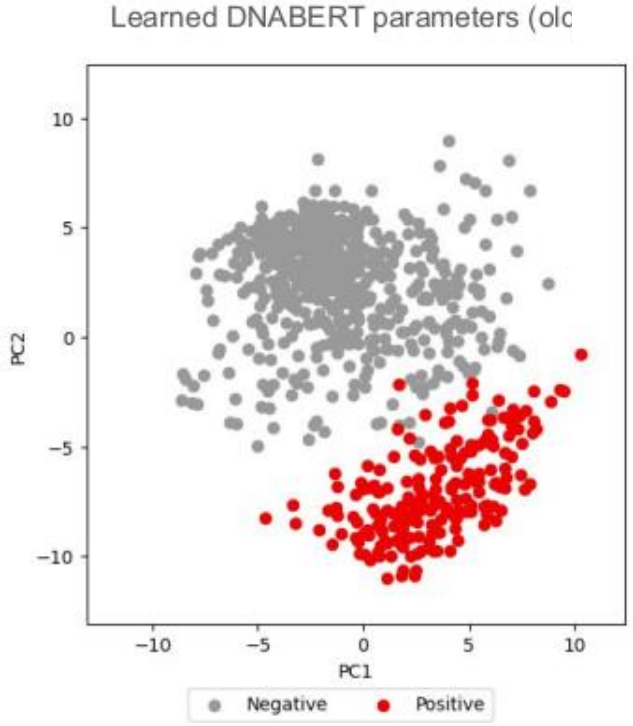
Figure 5: Examples of attribution trees. (a) is from MNLI, which is predicted as `entailment` by BERT. (b) is from SST-2, which is predicted as `positive` by BERT. The grey words from the inputs do not appear in the attribution trees.

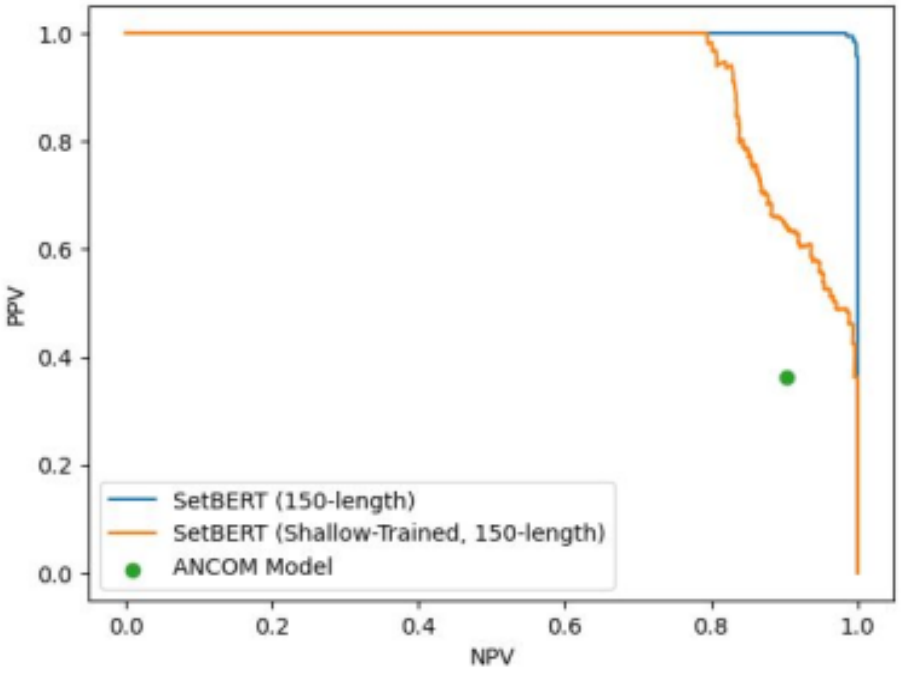
(b) Attribution Score

Hao, Yaru, et al. "Self-attention attribution: interpreting information interactions inside transformer." arXiv preprint arXiv:2004.11207 (2020).

SNAKE FUNGAL DISEASE (PATHOGEN PRESENCE/ABSENCE)

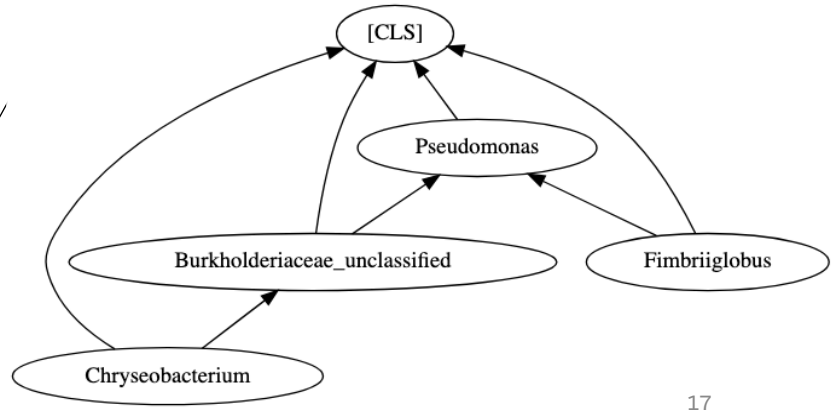






Improvements due to allowing gradient flow back into the DNABERT component of the SetBERT models. (Previous model used fixed DNABERT embeddings to preserve GPU memory).

Collaboration with Walker Lab at MTSU (Donald Walker, Department of Biology, MTSU)

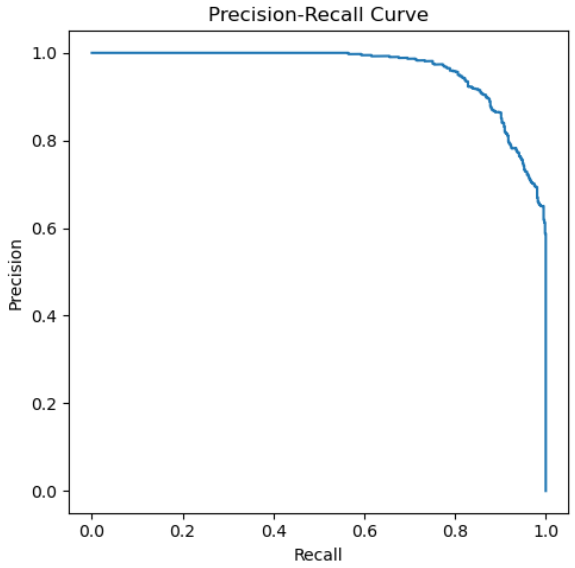


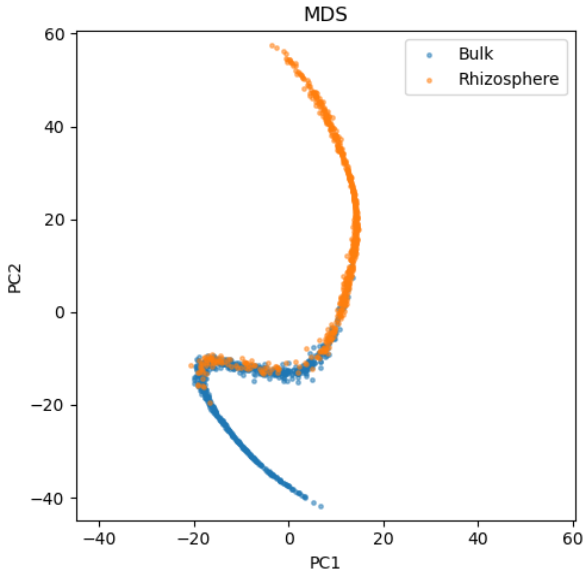
ATTENTION SHIFTS (PRETRAINED TO FINE-TUNED)

Predicted Pathogen Presence	Taxa Category	Phylum	Class	Order	Family	Genus	Score
Positive	Highest Scoring	Actinobacteria	Actinobacteria	Corynebacteriales	Dietziaceae	Dietzia	25,422
		Bacteroidetes	Bacteroidia	Flavobacteriales	Weeksellaceae	Chryseobacterium	25,348
		Proteobacteria	Alphaproteobacteria	Sphingomonadales	Sphingomonadaceae	Sphingomonas	22,800
		Firmicutes	Bacilli	Bacillales	Staphylococcaceae	Staphylococcus	22,000
		Proteobacteria	Betaproteobacteria	Burkholderiales	Comamonadaceae	Comamonas	21,500
		Proteobacteria	Gammaproteobacteria	Enterobacteriales	Enterobacteriaceae	Escherichia	21,001
		Firmicutes	Bacilli	Lactobacillales	Lactobacillaceae	Lactobacillus	20,500
		Bacteroidetes	Bacteroidia	Bacteroidales	Porphyromonadaceae	Parabacteroides	20,250
		Actinobacteria	Actinobacteria	Actinomycetales	Actinomycetaceae	Actinomyces	20,001
		Firmicutes	Clostridia	Clostridiales	Ruminococcaceae	Ruminococcus	19,850
	Lowest Scoring	Actinobacteria	Actinobacteria	Micrococcales	Dermacoccaceae	Flexivirga	-7,073
		Proteobacteria	Gammaproteobacteria	Enterobacteriales	Enterobacteriaceae	Enterobacteriaceae_unclassified	-6,149
		Bacteroidetes	Bacteroidia	Cytophagales	Microscillaceae	Siphonobacter	-4,966
		Proteobacteria	Gammaproteobacteria	Pseudomonadales	Moraxellaceae	Acinetobacter	-3,918
		Firmicutes	Bacilli	Lactobacillales	Enterococcaceae	Enterococcus	-3,339
		Actinobacteria	Actinobacteria	Micrococcales	Dermabacteraceae	Brachybacterium	-2,196
		Proteobacteria	Gammaproteobacteria	Pseudomonadales	Pseudomonadaceae	Pseudomonas	-2,130
		Firmicutes	Bacilli	Bacillales	Bacillales_unclassified	Bacillales_unclassified	-1,825
		Firmicutes	Erysipelotrichia	Erysipelotrichales	Erysipelotrichaceae	Erysipelothrix	-1,765
		Firmicutes	Bacilli	Bacillales	Staphylococcaceae	Staphylococcus	-1,346

Collaboration with Kateryna Zhalnina
(Lawrence-Berkeley National Laboratory)

BULK SOIL VS. RHIZOSPHERE DISTINCTIONS



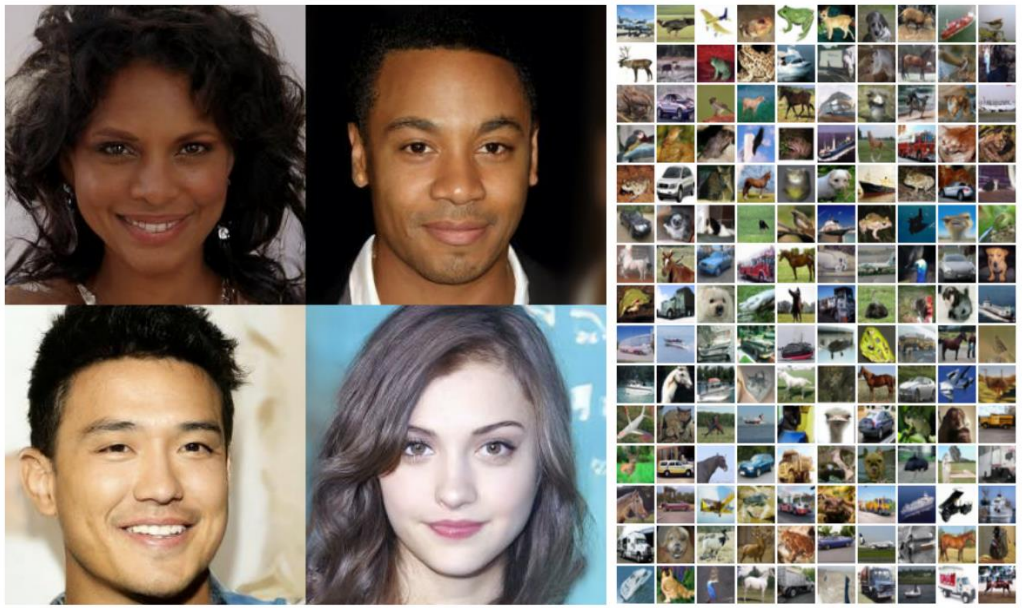


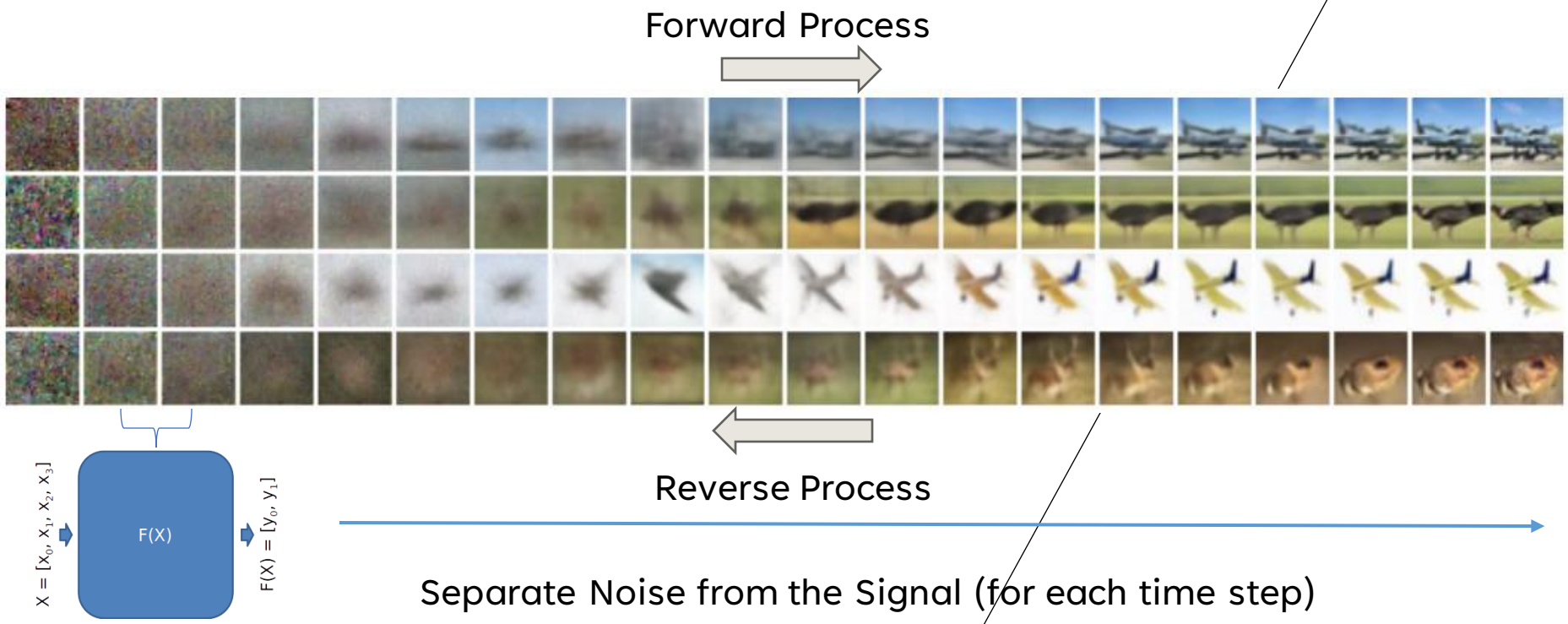
Actinobacteriota	Thermoleophilia	Solirubrobacterales
Acidobacteriota	Vicinamibacteria	Vicinamibacterales
Planctomycetota	Planctomycetes	Gemmatales
Actinobacteriota	Thermoleophilia	Solirubrobacterales
Gemmatimonadota	Gemmatimonadetes	Gemmatimonadales
Proteobacteria	Alphaproteobacteria	Rhizobiales
Acidobacteriota	Thermoanaerobaculia	Thermoanaerobaculales
Proteobacteria	Alphaproteobacteria	Rhizobiales
Actinobacteriota	Thermoleophilia	Gaiellales
Acidobacteriota	Vicinamibacteria	Vicinamibacterales

Firmicutes	Bacilli	Bacillales
Firmicutes	Bacilli	Bacillales
Firmicutes	Bacilli	Bacillales
Firmicutes	Bacilli	Bacillales
Proteobacteria	Alphaproteobacteria	Sphingomonadales
Bacteroidota	Bacteroidia	Chitinophagales
Verrucomicrobiota	Verrucomicrobiae	Chthoniobacterales
Firmicutes	Bacilli	Bacillales
Bacteroidota	Bacteroidia	Chitinophagales
Bacteroidota	Bacteroidia	Chitinophagales

DIFFUSION MODELS

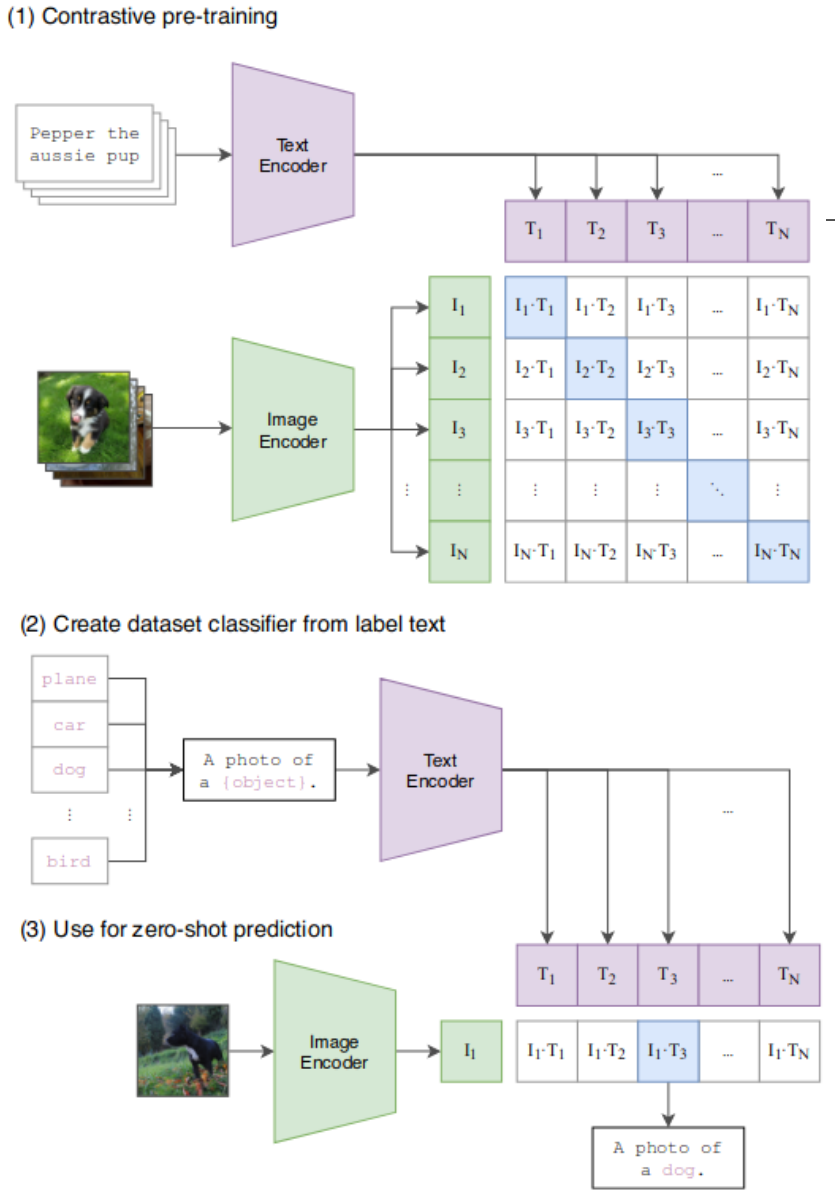
- Ho et al., 2020
- Denoising Diffusion Probabilistic Models (DDPM)
- An elegant solution to the *mode collaspe* issue with generative modeling tasks





CONTRASTIVE EMBEDDING ALIGNMENT ACROSS DOMAINS

- Radford et al., 2021
- Contrastive learning can be done to align the learned information from one domain to another related domain
- Image to text
- Text to image
- Doesn't have to just be these domains...
- Zero-shot prediction still possible, just like with the earlier SimCLR approach



CONDITIONAL GENERATION WITH CONTRASTIVE EMBEDDINGS: DALL-E

- Ramesh et al., 2022
- Reverse diffusion process can be trained while including a *contrastive embedding*
- The text encoder can generate a *similar* contrastive embedding
- The diffusion model can use the text's contrastive embedding to extract a *similar* image
- Natural variation in the DDPM model and the large variation in training data allows for creative, generative modeling

vibrant portrait painting of Salvador Dalí with a robotic half face

a shiba inu wearing a beret and black turtleneck

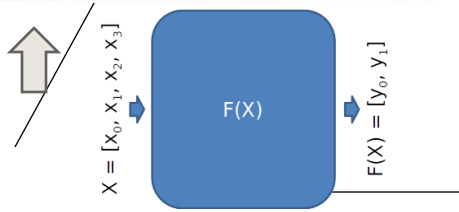
a close up of a handpalm with leaves growing from it

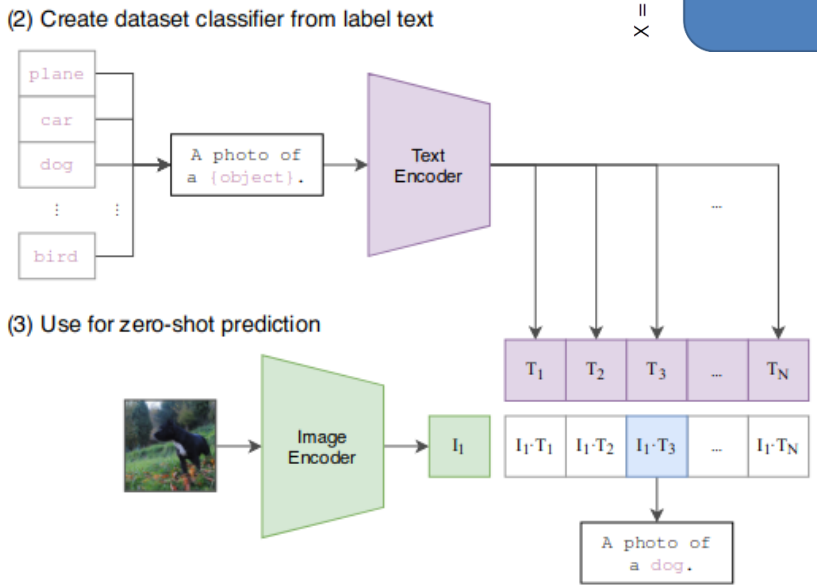
an espresso machine that makes coffee from human souls, artstation

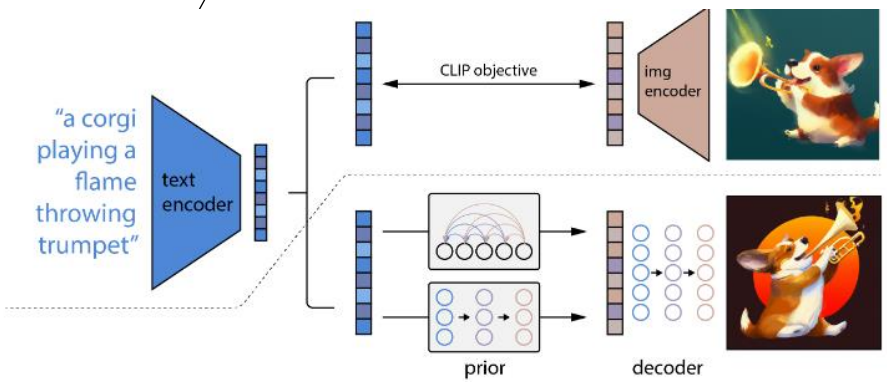
panda mad scientist mixing sparkling chemicals, artstation

a corgi's head depicted as an explosion of a nebula

Conditioned Diffusion







SOME FINAL THOUGHTS AND TAKE-AWAYS

1. Language, images, audio (not covered here, but improvements in this domain are steady now as well): *all human communication* and therefore all domains of human knowledge are directly impacted by Generative AI.
2. Recent work has shown they often out-perform humans on abstract reasoning tasks (Webb et al., Aug. 2023). "*Our results indicate that large language models such as GPT-3 have acquired an emergent ability to find zero-shot solutions to a broad range of analogy problems.*"
3. Entire scientific process pipelines are being assembled using multiple coordinating generative models: hypothesis generation, experimental design, search and execution, and analysis/revision.
4. What about Quantum?
    1. Some functions of interest rely on quantum phenomena and AI may soon be used to learn these quantum functions instead of relying on human programming.
    2. Better optimization and approximation may be possible by integrating quantum algorithms into AI systems and this is an active area of research.

THANK YOU

Joshua L. Phillips

Joshua.Phillips@mtsu.edu

https://phillips-lab.org/