Contextures:

Mechanism of Representation Learning

Runtian Zhai

Computer Science Department

Carnegie Mellon University

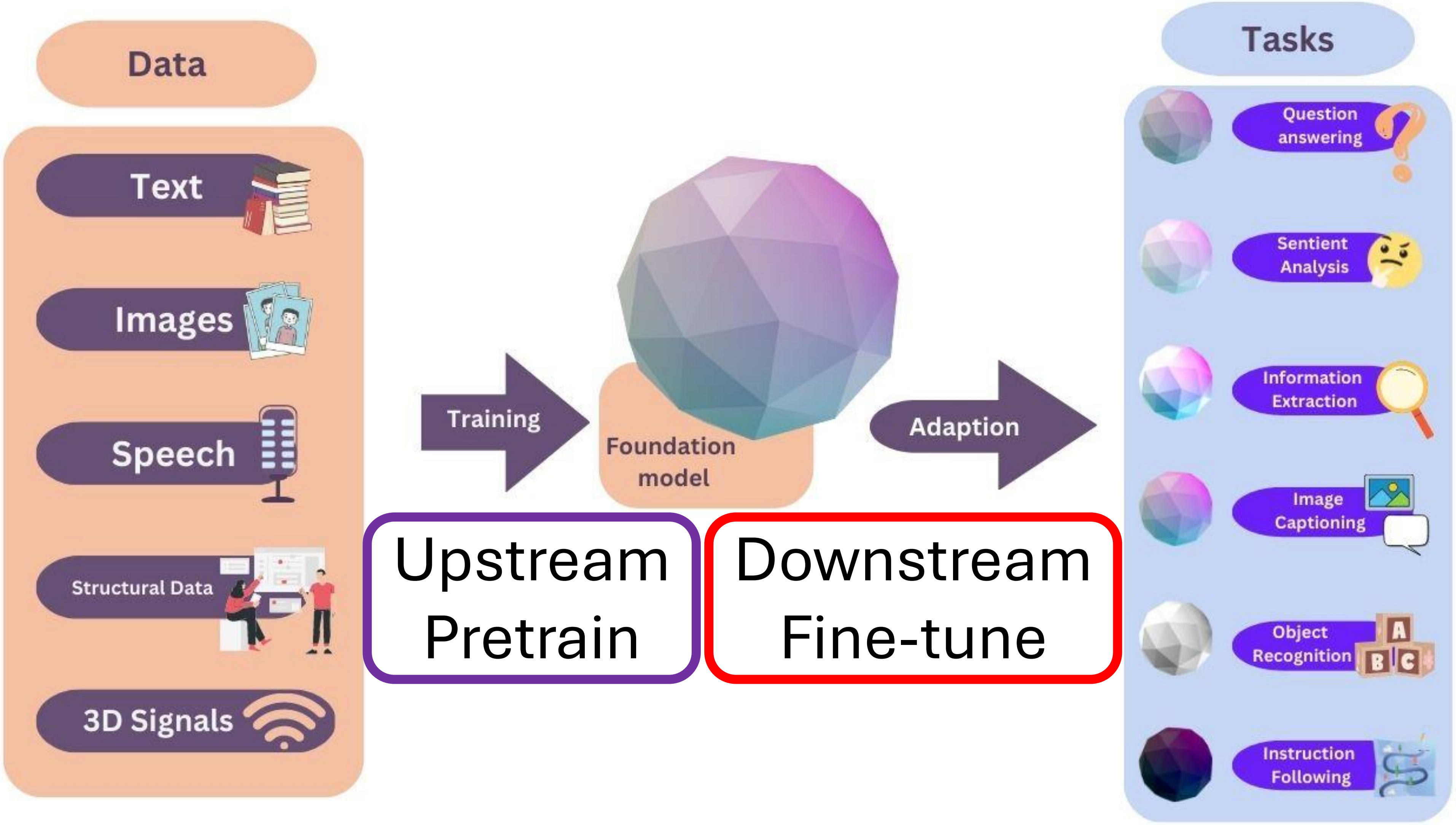
April 15, 2025

Carnegie Mellon University

New Machine Learning Paradigm

End-to-end Deep Learning → Foundation Models

Representation Learning & Foundation Models



Huge Empirical Success

ChatGPT

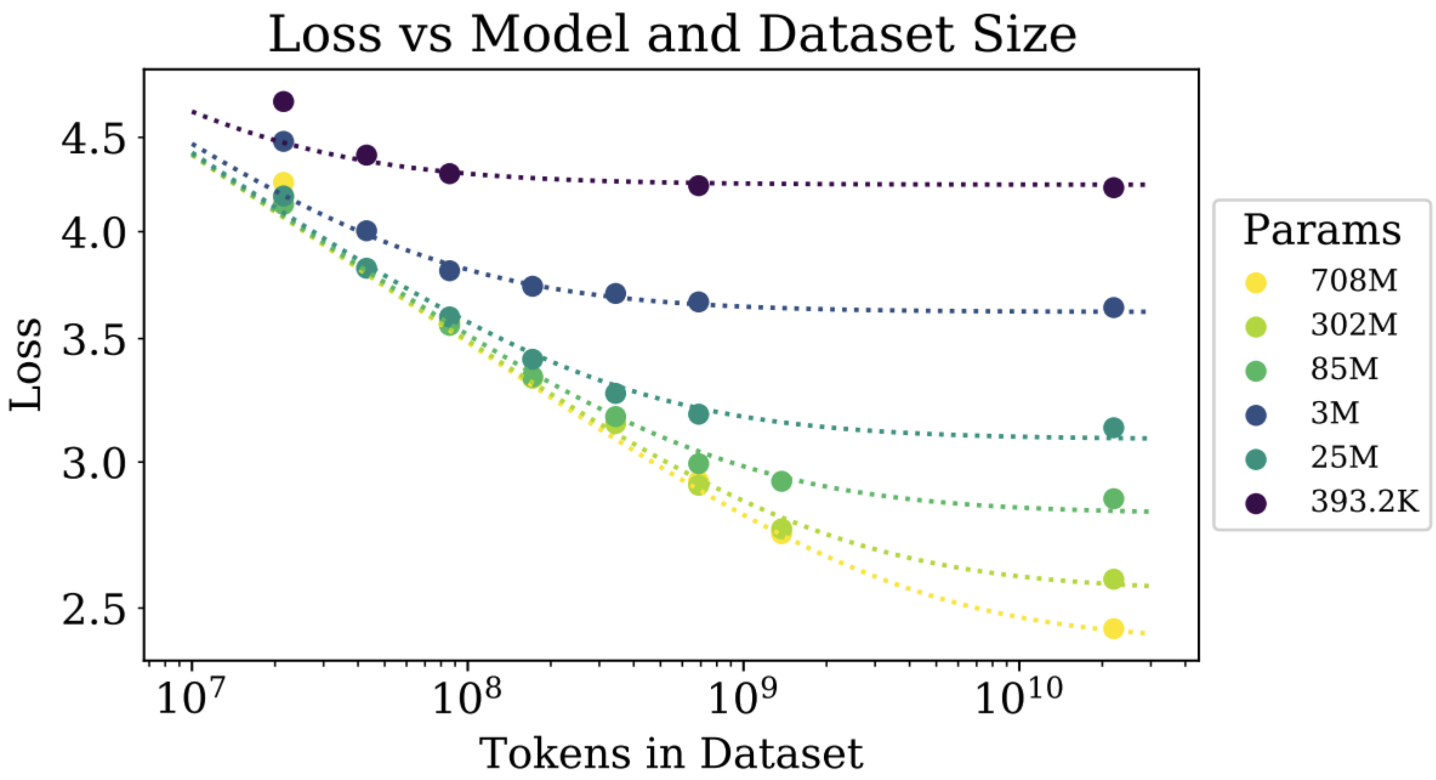
deepseek

AI Claude 3

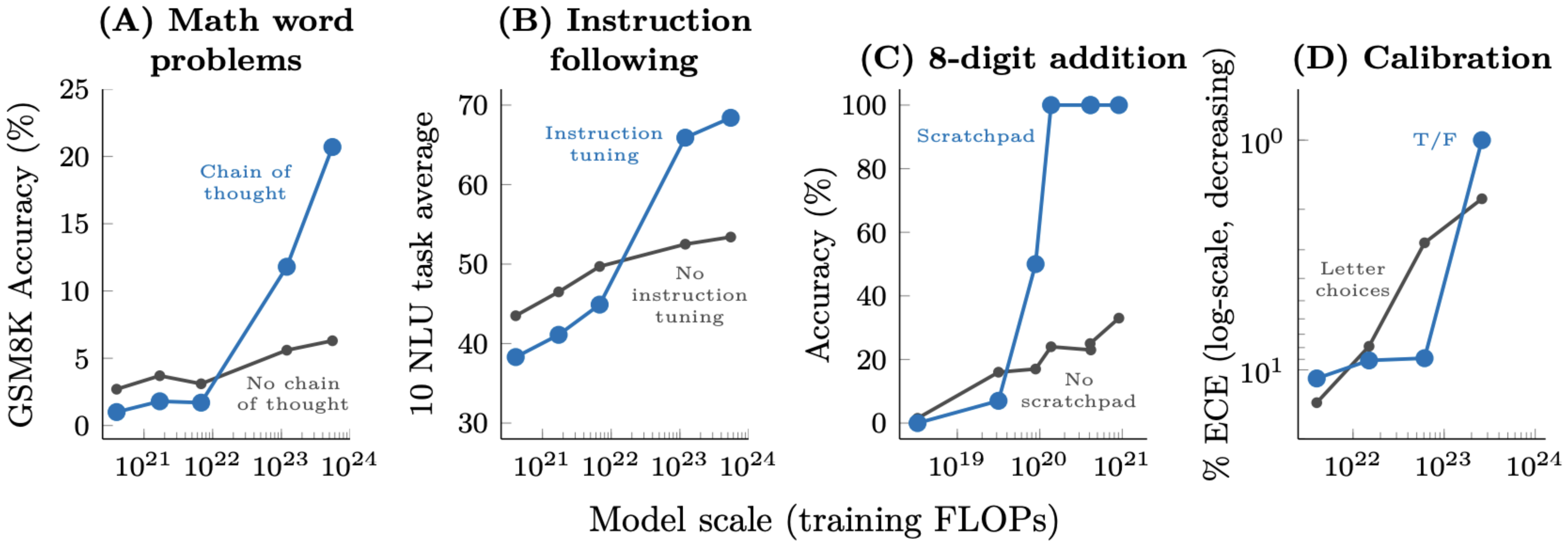
LLaMA

Scaling Law

- Larger models and larger datasets lead to better performance



Emergent Abilities



Wei et al. Emergent Abilities of Large Language Models

Carnegie Mellon University

Current Situation

Carnegie Mellon University

Current Situation

Home → Tech → AI

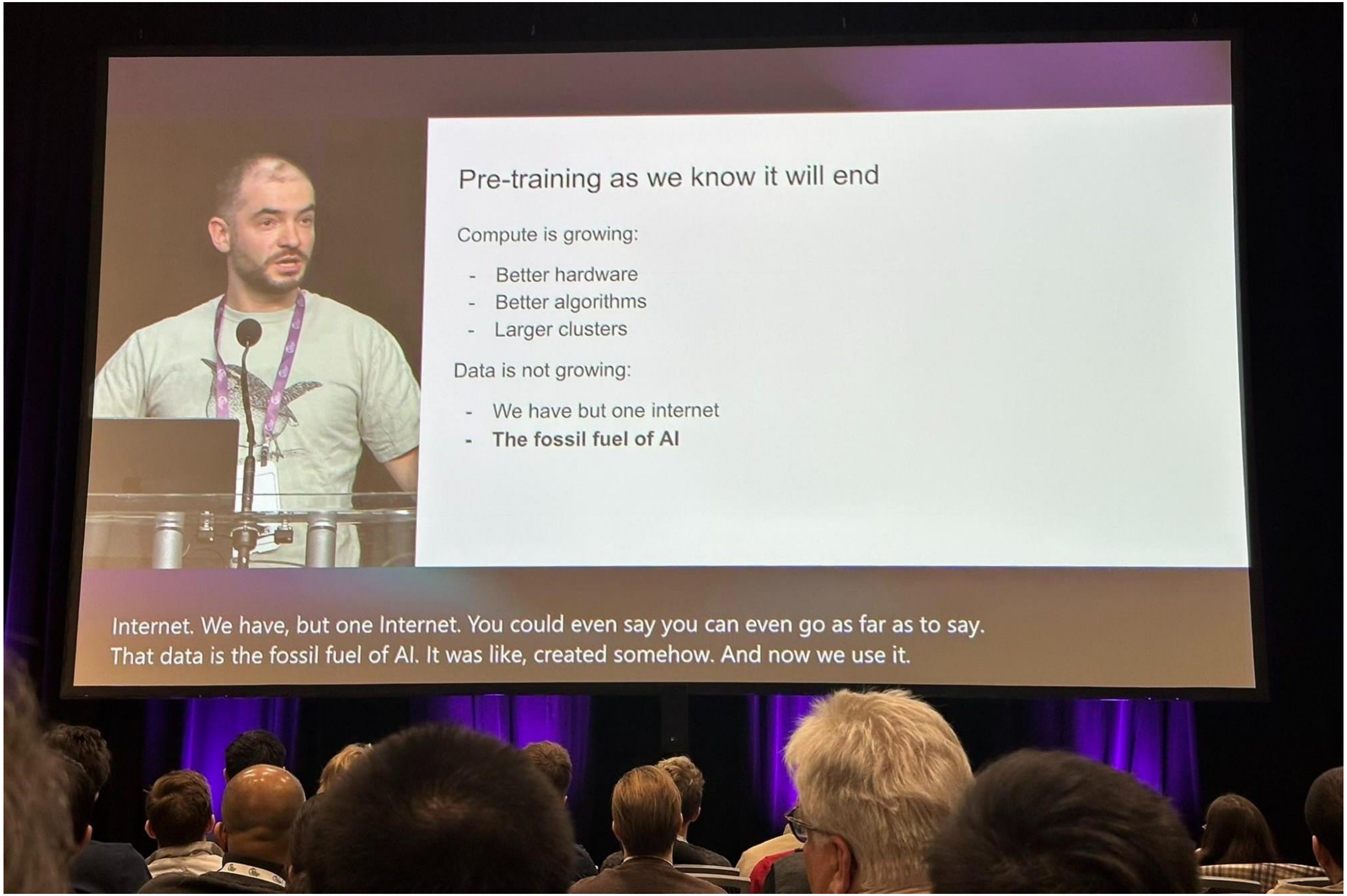
I Tested Grok 3, and It's Not Worth the Price Hike

You can get a lot more for less.

r/singularity • 21 days ago
Neurogence Top 1% Commenter

Grok 3 Not Performing Well In Real World Performance: What Does This Say About Benchmarks And Scaling?

Carnegie Mellon University



Carnegie Mellon University

Scaling Law is Not All We Need

- The scaling law is only an empirical “law”
- Like Moore’s law or any other law, it’s not always reliable

We need to have a scientific understanding of the mechanism of representation learning (pretraining)!

Without such an understanding…

- We don’t know the limits of scaling
- We don’t know when to expect those limits
- We don’t know how to advance once it reaches the limits

This thesis

- Establishes the contexture theory
- Which delineates what representation a foundation model learns
- Implies that making models larger has diminishing returns
- And shows how to make further progress beyond scaling

Central Argument:
Representations are learned from the association between the input X and a context variable A

This is called the learning the contexture

A Unified Theory

- Prior work treats different pretraining methods in very different ways
- The contexture theory unifies a variety of methods, such as
  - Supervised learning
  - Contrastive learning
  - Masked autoencoders
  - Manifold learning
  - Kernel machines
  - Diffusion models
  - Knowledge distillation

Carnegie Mellon University

This talk

- Part 1: Fundamentals of the contexture theory
- Part 2: How to learn the contexture?
- Part 3: Learning the contexture is optimal
- Part 4: How to obtain better contexts?

Part 1: Fundamentals

Problem Setting

- **Data:** Unlabeled samples $\gg$ Labeled samples
- **Pretraining:** Learn a d-dimensional encoder $\Phi: \mathcal{X} \to \mathbb{R}^d$ using the unlabeled samples
- **Downstream:** Fit a linear model using the labeled samples

$$\hat{f}(x) = W\Phi(x) + b$$

Linear probe

Carnegie Mellon University

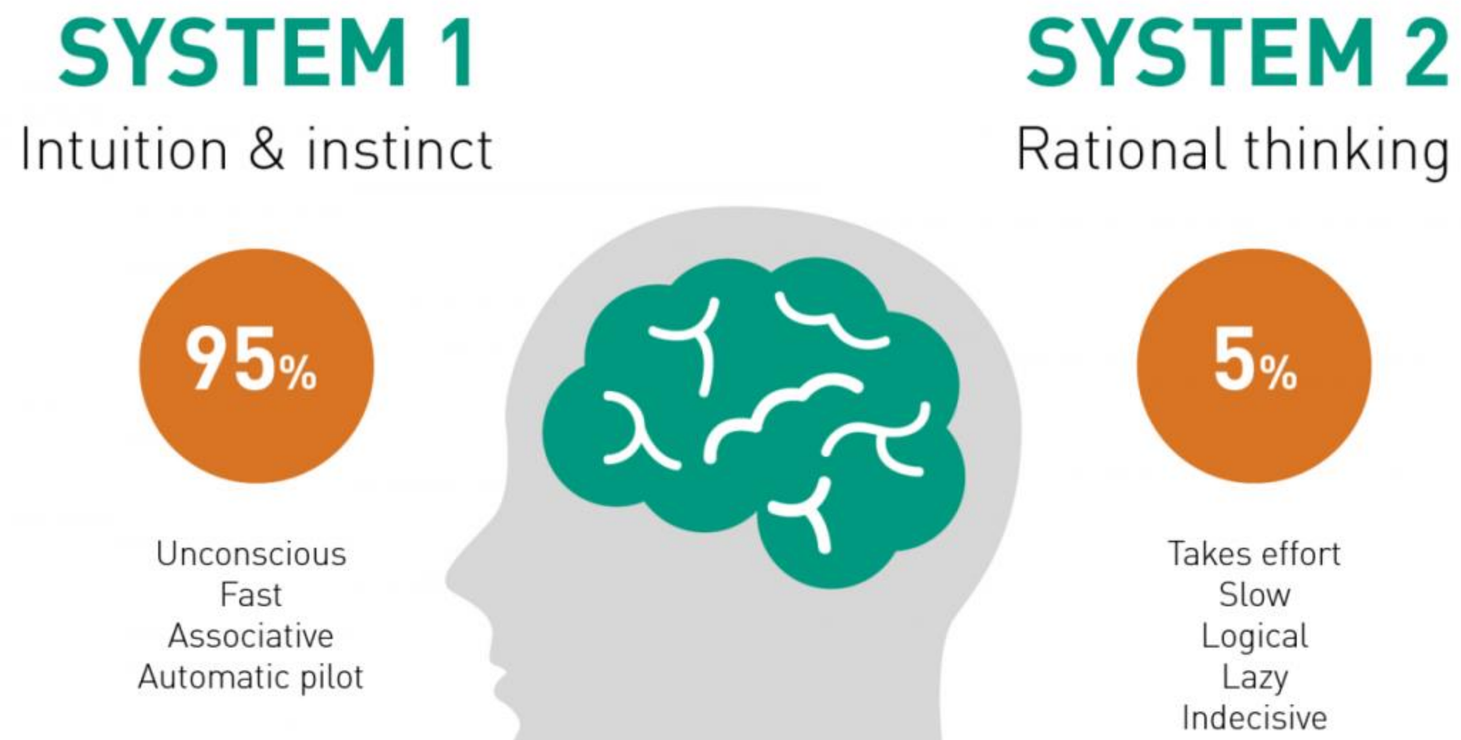
Result #1:

Representations are learned from the association between the input X and a context variable A

This association is called the contexture

Two systems of thinking by Daniel Kahneman

Representation learning. It is fast and associative, but it is bad at reasoning



- Recognizing an image of a crosswalk
- Calculating 2 + 2
- Which state is Pittsburgh in?

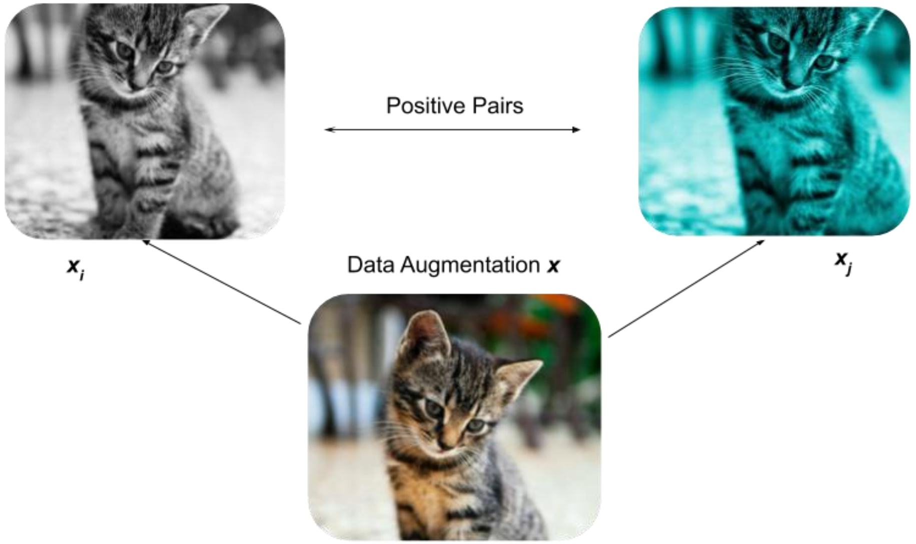
- ReCAPTCHA: Select all crosswalks
- Calculating 177×284
- How many states start with “N”?

Ilya Sutskever’s Deep Learning Hypothesis

- **“If you have a large neural network, it can do anything a human can do in a fraction of a second.”**
- Representation learning can do any System 1 thinking
  - Better than humans thanks to its larger memory and faster compute

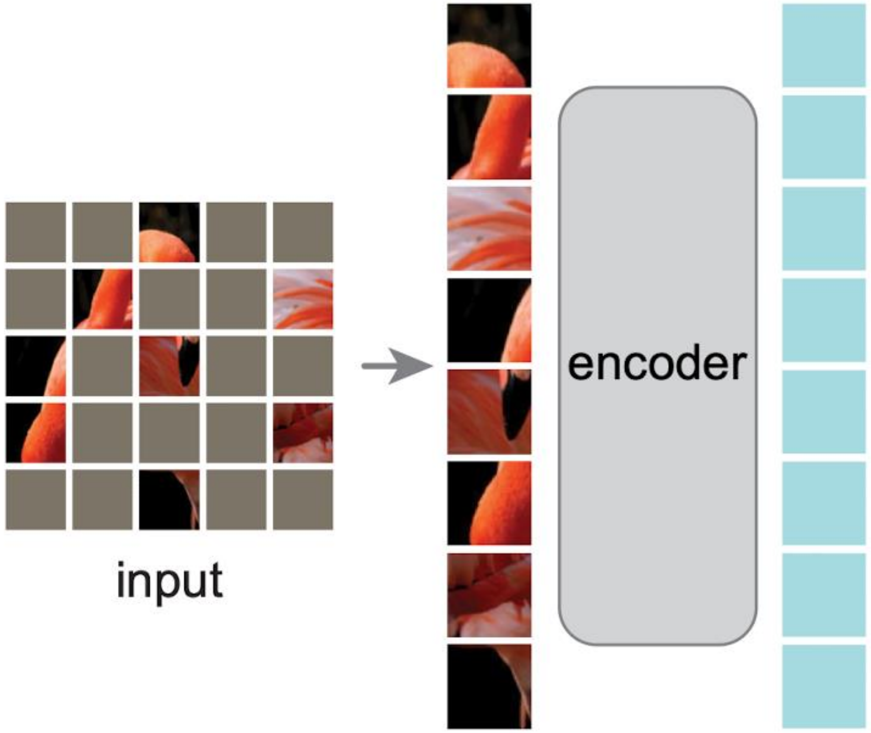
Example: Self-supervised Learning

Contrastive learning



X = Image   A = View (corrupted image)

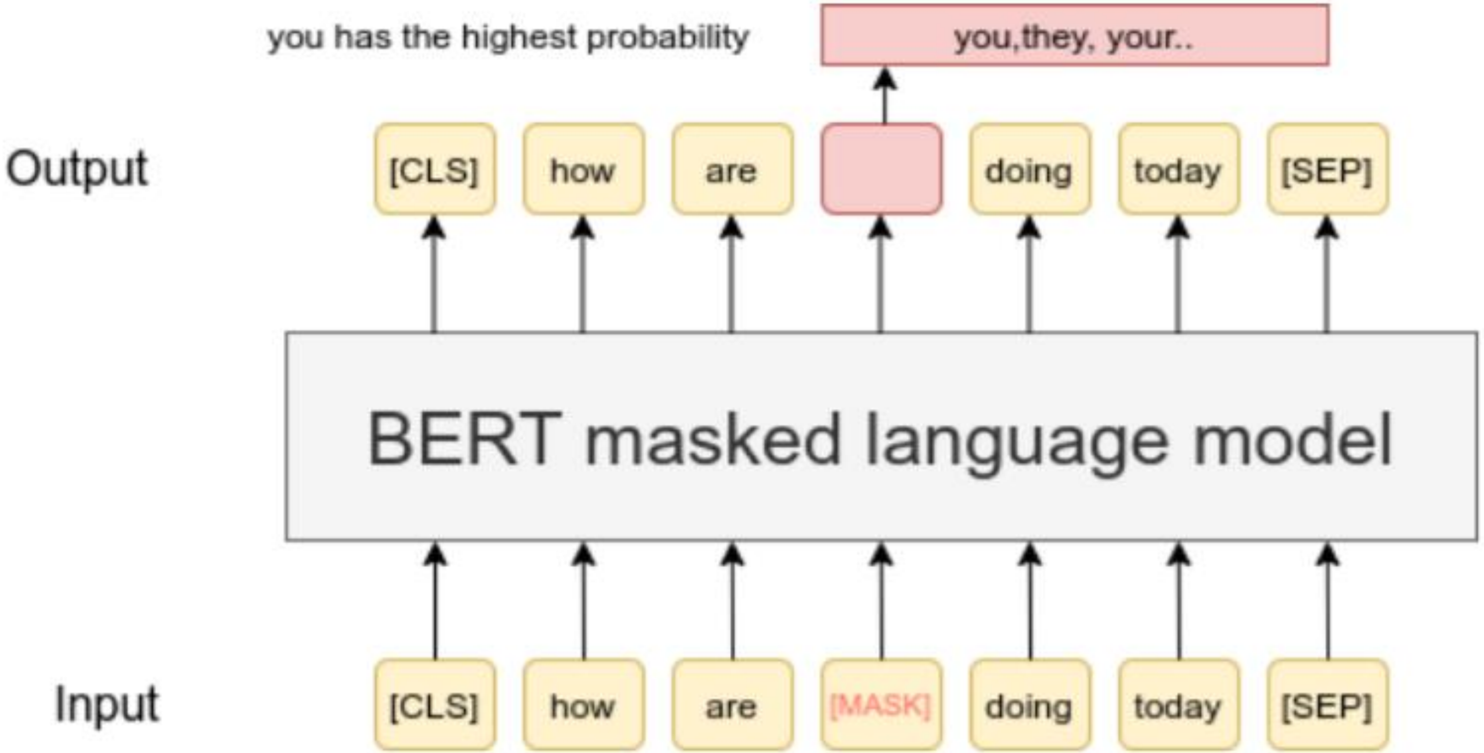
Masked autoencoders



X = Image   A = Masked image

Carnegie Mellon University

Example: BERT

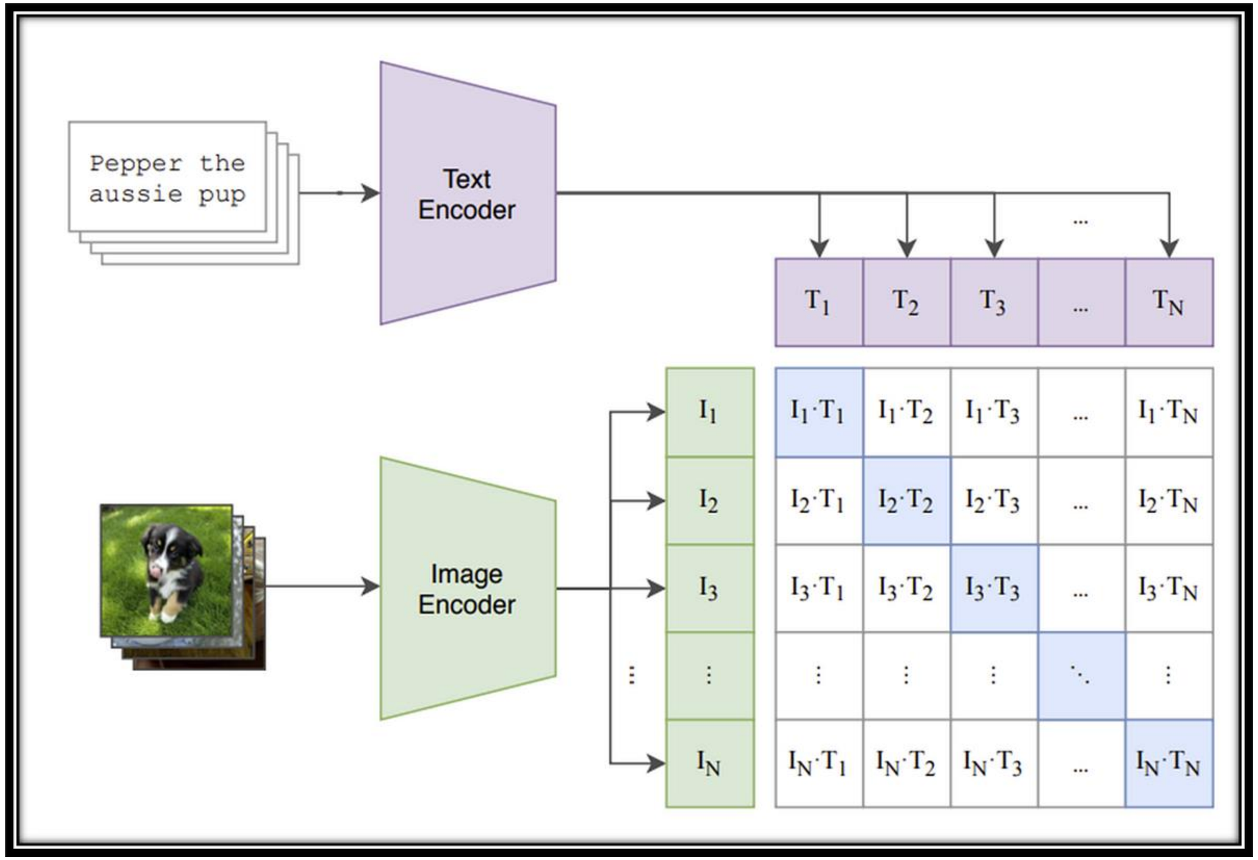


X = Text A = Masked text

Image from sbert.net

Example: CLIP

- Vision-language model



X = Image
A = Text caption

Carnegie Mellon University

Context Variable

Method	Input X	Context Variable A
Supervised learning	Sample	Label
K-nearest neighbor	Sample	Neighbor of X
Diffusion models	Image	X plus noise
Contrastive learning	Image	A view of the image
BERT	Text	Masked text
Vision-language models (CLIP)	Image	Text caption

Context Variable

- **Joint distribution** $P^{+}(X, A)$
  - With marginals P_X and P_A
- $\boldsymbol{L^2}$ **function spaces** $L^2(P_X), L^2(P_A)$:

$$\langle f_1, f_2 \rangle_{P_X} = \mathbb{E}_{P_X}[f_1(X) f_2(X)]$$
$$\langle g_1, g_2 \rangle_{P_A} = \mathbb{E}_{P_A}[g_1(A) g_2(A)]$$

- **Expectation operator** $T_{P^+}: L^2(P_A) \to L^2(P_X)$

$$(T_{P^+} g)(x) = \int g(a) P^{+}(a|x) da = \mathbb{E}[g(A)|x]$$

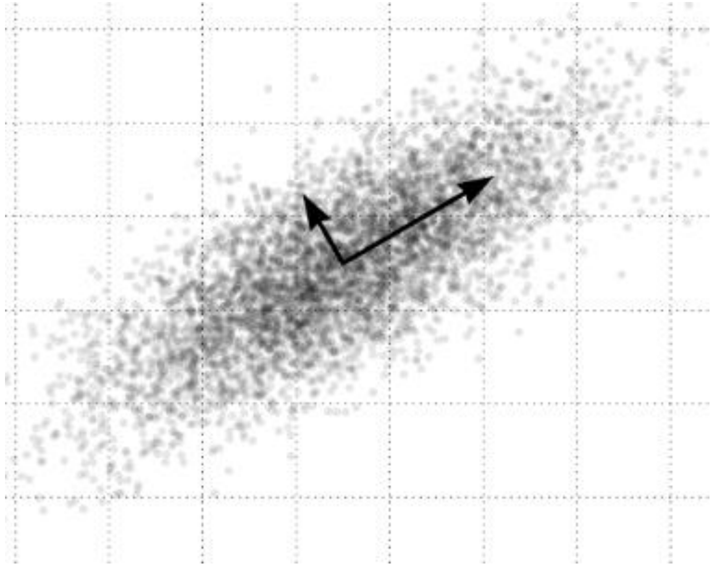
Carnegie Mellon University

Result #2:
Representation learning preserves
the most information of T_{P^+}

We call this "learning the contexture"

Carnegie Mellon University

Analogy: PCA



- Finite spaces $|\mathcal{X}| = N, |\mathcal{A}| = M$
- $f \in L^2(P_X), g \in L^2(P_A)$ are vectors in $\mathbb{R}^N, \mathbb{R}^M$
- T_{P^+} is a matrix $T \in \mathbb{R}^{N \times M}$
- Goal: Learn a d-dim embedding $E \in \mathbb{R}^{N \times d}$
- PCA: If E is the top-d left singular vectors of T, then it maximizes the explained variance (information)

SVD of T_{P^+}

- $s_i \in \mathbb{R}$: Singular values. $1 = s_0 \geq s_1 \geq \cdots \geq 0$
- $\mu_i \in L^2(P_X), \nu_i \in L^2(P_A)$: Left/right singular functions
- Zeroth singular functions are constant: $\mu_0 \equiv \nu_0 \equiv 1$
- Dual kernel integral operator: $T_{k_X^+} = T_{P^+} T_{P^+}^*$
  - Similar to $TT^\top$ in the vector space scenario
- (s_i^2, μ_i) are the eigenvalues and eigenfunctions of $T_{k_X^+}$

Carnegie Mellon University

Learning the Contexture

- An encoder $\Phi = [\phi_1, \cdots, \phi_d]$ is said to learn the contexture, if $\mathrm{span}(\phi_1, \cdots, \phi_d) = \mathrm{span}(\mu_1, \cdots, \mu_d)$
- Also called extracting the top-d eigenspace of $T_{k_X^+}$
- **Remark 1:** It only recovers the linear span, but does not recover the exact function μ_i (which is much harder)
- The downstream model is linear, so we only care about the span

Carnegie Mellon University

Learning the Contexture

- An encoder $\Phi = [\phi_1, \cdots, \phi_d]$ is said to learn the contexture, if $\operatorname{span}(\phi_1, \cdots, \phi_d) = \operatorname{span}(\mu_1, \cdots, \mu_d)$
- Also called extracting the top-d eigenspace of $T_{k_X^+}$
- **Remark 2:** It excludes $\mu_0 \equiv 1$
- The bias term b in downstream linear model implicitly includes μ_0

Learning the Contexture

- An encoder $\Phi = [\phi_1, \cdots, \phi_d]$ is said to learn the contexture, if $\mathrm{span}(\phi_1, \cdots, \phi_d) = \mathrm{span}(\mu_1, \cdots, \mu_d)$
- Also called extracting the top-d eigenspace of $T_{k_X^+}$

Learning the contexture =
Learning a d-dimensional space

Next

- Part 2: How to learn the contexture?
- Part 3: Why learning the contexture is optimal?

Part 2:
How to learn the contexture?

Classical Method: Kernel PCA

- m pretraining samples
- Eigen-decompose an $m \times m$ Gram matrix
- Time complexity: About $O(m^3)$
- Not scalable when m is huge

Result #3:

Contextures can be learned by training a large model to optimize certain variational objectives

The “deep learning” way

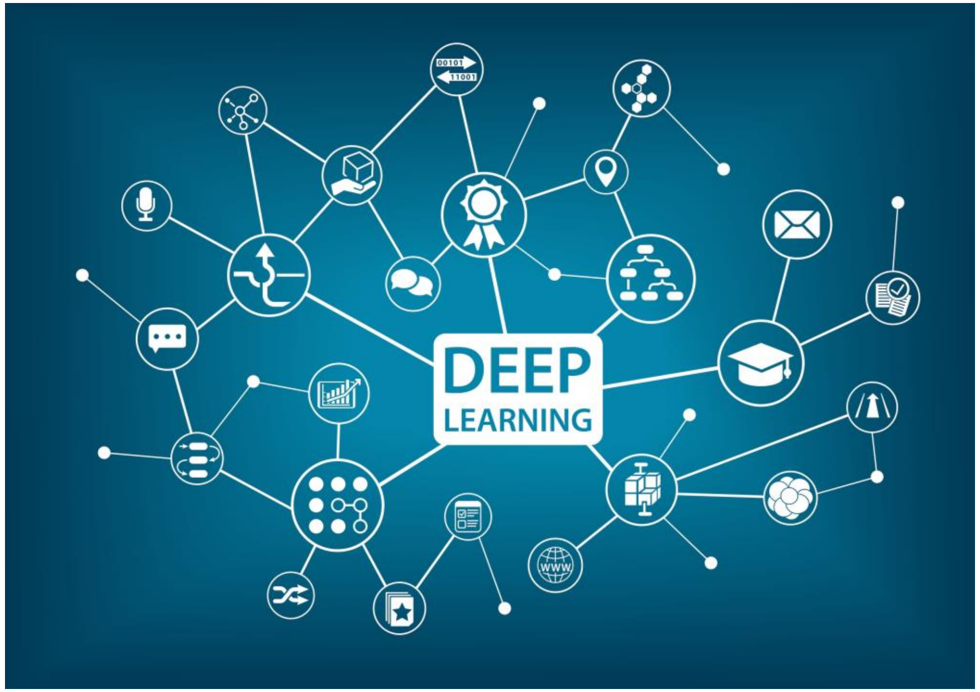
Variational Objective $\mathcal{R}(\Phi)$

- Condition: The minimizer of $R(\Phi)$ on $L^2(P_X)$ is the Φ^* that learns the contexture
- Then, it suffices to have an expressive model and a good optimizer

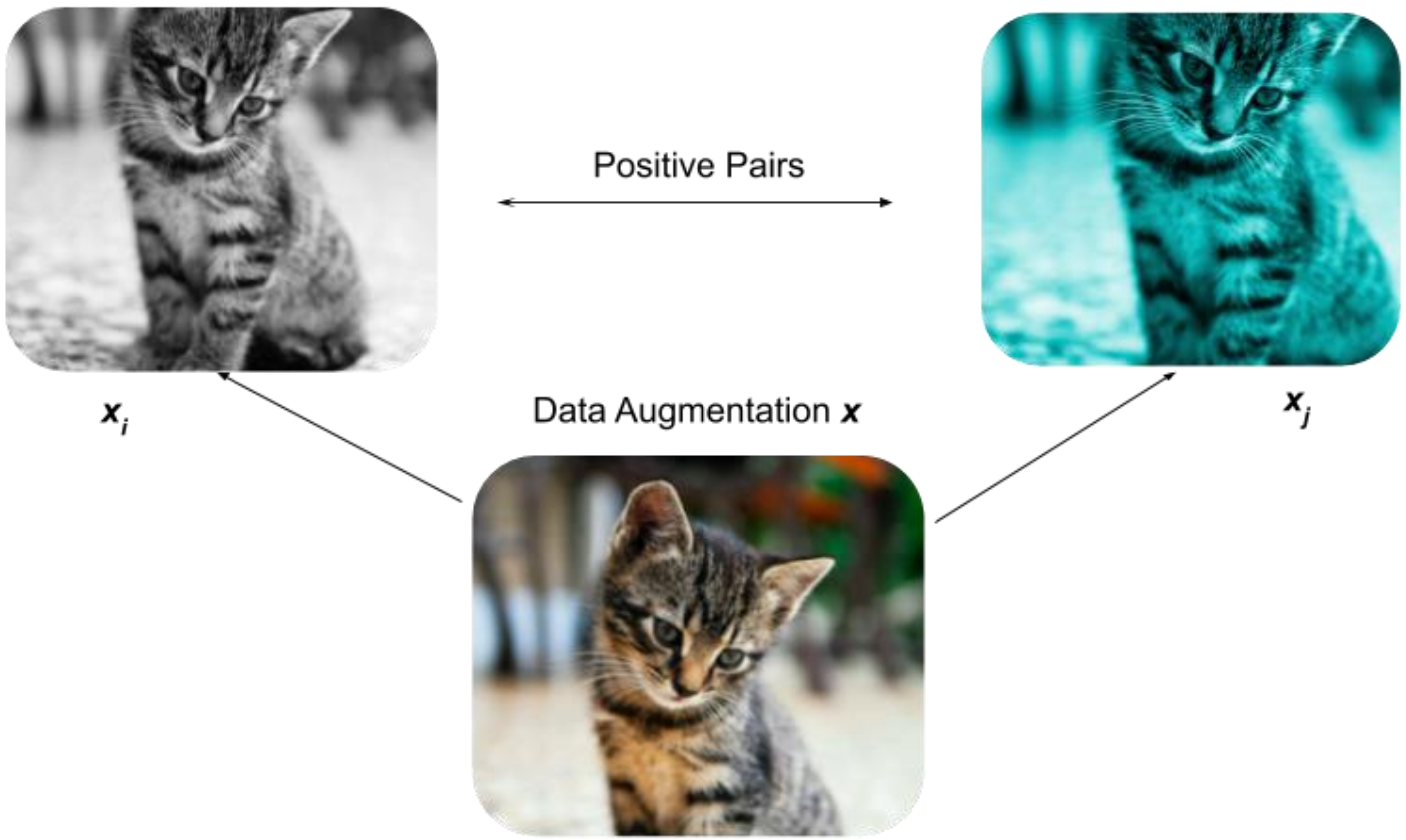
Carnegie Mellon University

This method works for

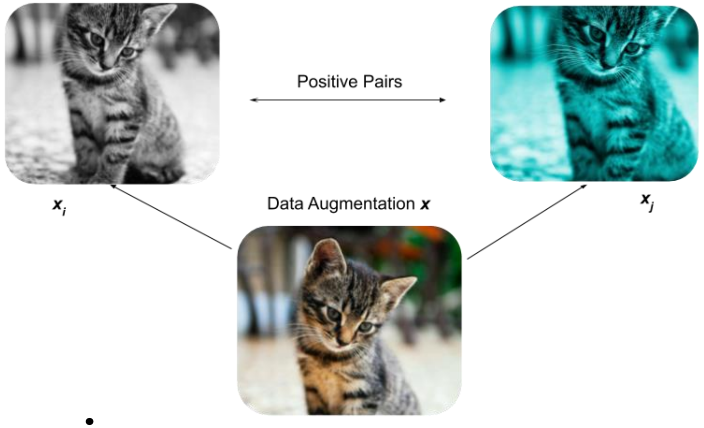
- Supervised learning
- Multi-view learning
- Masked autoencoders
- Manifold learning
- Kernel machines
- Diffusion models
- Knowledge distillation



Example: Multi-view Learning



Multi-view Learning



- Positive pair: $A, A^+ \sim P^+(\cdot\,|X)$
- Goal: Give similar embeddings to a positive pair
- Degenerate solution: Same embedding to all A (**feature collapse**)
- **Non-contrastive learning**: Train $\Psi: \mathcal{A} \to \mathbb{R}^d$ to

$$\text{Minimize} \quad \mathbb{E}_{X \sim P_X}\left[\mathbb{E}_{A,A^+ \sim P^+(\cdot|X)}[\|\Psi(A) - \Psi(A^+)\|_2^2]\right]$$

$$\text{s.t.} \quad \mathrm{Cov}_{P_A}[\Psi] = I \quad \text{(orthonormality constraint)}$$

- Ψ must be rank-d, so cannot be constant on all A

Carnegie Mellon University

Multi-view Learning

$$\text{Minimize} \quad \mathbb{E}_{X \sim P_X}\left[\mathbb{E}_{A,A^+ \sim P^+(\cdot|X)}[\|\Psi(A) - \Psi(A^+)\|_2^2]\right]$$
$$\text{s.t.} \quad \mathrm{Cov}_{P_A}[\Psi] = I \quad \text{(orthonormality constraint)}$$

- The learned encoder is Ψ for A, not Φ for X
- Conversion: **Average encoder** $\Phi(x) = (T_{P^+}\Psi)(x) = \mathbb{E}[\Psi(A)|x]$
- **Theorem:** If Ψ^* is the optimal solution, then Φ^* learns the contexture

Carnegie Mellon University

All of these learn the contexture

- Supervised learning
- Multi-view learning
- Masked autoencoders
- Manifold learning
- Kernel machines
- Diffusion models
- Knowledge distillation

Many existing pretraining objectives can learn the contexture.

But why is the contexture good?

Part 3:
Learning the Contexture is Optimal

Carnegie Mellon University

“Optimal”?

- Optimal means it leads to the best encoder Φ
- We define “best”, by defining how Φ is evaluated

How to evaluate an encoder Φ?

- **Extrinsic evaluation:** Performance of Φ on a specific task
  - This is what we ultimately care about
- **Intrinsic evaluation:** Evaluate Φ without any task
  - We might not know all the tasks at pretrain time
  - We want Φ to be transferable to new tasks

Intrinsic Evaluation

- No model is good on all tasks, but not all tasks are important
- Counting how many words in a paragraph contain the letter "e" is a valid task. LLMs are bad at it, but most people don't care

Carnegie Mellon University

Example

RZ Count how many words in the following paragraph contains the letter "e":
Overall, although there does not exist a universal metric that works for all contexts and tasks, and our metric does have failure cases, the experiment results here provide empirical evidence that more often than not, the proposed metric correlates well with the actual prediction error of the downstream linear probe. Hence, the proposed metric is useful for choosing hyperparameters and comparing contexts in practice.

I'll count the words containing the letter "e" in the paragraph you provided. Let me go through it word by word:

1. "Overall" - contains "e"
2. "although" - contains "e"
3. "there" - contains "e"
4. "does" - contains "e"
5. "not" - no "e"
6. "exist" - contains "e"
7. "a" - no "e"
8. "universal" - contains "e"
9. "metric" - contains "e"
10. "that" - no "e"
11. "works" - no "e"
12. "for" - no "e"
13. "all" - no "e"
14. "contexts" - contains "e"
15. "and" - no "e"
16. "our" - no "e"

Example

- **Runtian:**

Count how many words in the following paragraph contains the letter “e”:

Overall, although there … and comparing contexts in practice.

- **Claude 3.7 Sonnet on April 14, 2025:**

I'll count the words containing the letter "e" in the paragraph you provided. Let me go through it word by word:

1. "Overall" - contains “e” …

59. "comparing" - contains “e” **(wrong)**

60. "contexts" - contains “e” …

(final answer is also wrong)

We only care about a small set of tasks

- This set can be defined with the context
- Prior knowledge: The context is useful for the task
  - We say that the context and the task are compatible
- Intrinsic evaluation: Evaluate Φ on all compatible tasks

Result #4:
No encoder is good for all tasks.
But contexture is optimal for tasks that are **compatible** with the context

Compatibility

- Task target function $f^*: \mathcal{X} \to \mathbb{R}$
- A context P^+ and a task f^* are compatible, if the context is useful for learning a predictor for the task
- Formally: They are compatible if a one can learn a predictor for f^* using the corrupted dataset $\{(a_1, f^*(x_1)), \cdots, (a_m, f^*(x_m))\}$
    - We don't see original x_i but can only see one $a_i \sim P^+(\cdot \mid x_i)$

Compatibility

- The compatibility between task f^* and context P^+ is

$$\rho(f^*, P^+) = \max_{g \neq 0} \frac{\langle f^*, T_{P^+} g \rangle}{\|f^*\|_{P_X} \|g\|_{P_A}} \in [0,1]$$

- Theorem: If $\rho(f^*, P^+)$ is close to 1, then a predictor for f^* can be learned using $\left\{\left(a_1, f^*(x_1)\right), \cdots, \left(a_m, f^*(x_m)\right)\right\}$

Intrinsic Evaluation of Φ

- Class of compatible tasks:

$$\mathcal{F}_\epsilon = \{f^*: \rho(f^*, P^+) \geq 1 - \epsilon\}$$

- Worst-case approximation error:

$$err(\Phi; \mathcal{F}_\epsilon) = \max_{f^* \in \mathcal{F}_\epsilon, \|f^*\|_{P_X} = 1} \min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \|w^\top \Phi + b - f^*\|_{P_X}^2$$

Optimal linear model on Φ for f^*

- **Theorem:** Among all d-dim encoders, the Φ that learns the contexture is a minimizer of $err(\Phi; \mathcal{F}_\epsilon)$, so it is optimal

Carnegie Mellon University

Learning the contexture is optimal if the task is known to be compatible with the context!

Result #5:

Increasing model size has diminishing returns! Further improvement requires better contexts

Intuition

- For a larger model, function class of $\Phi \to L^2(P_X)$
- Optimizer of $\mathcal{R}(\Phi) \to$ top-d eigenfunctions
- When they are close enough, further scaling has little benefit

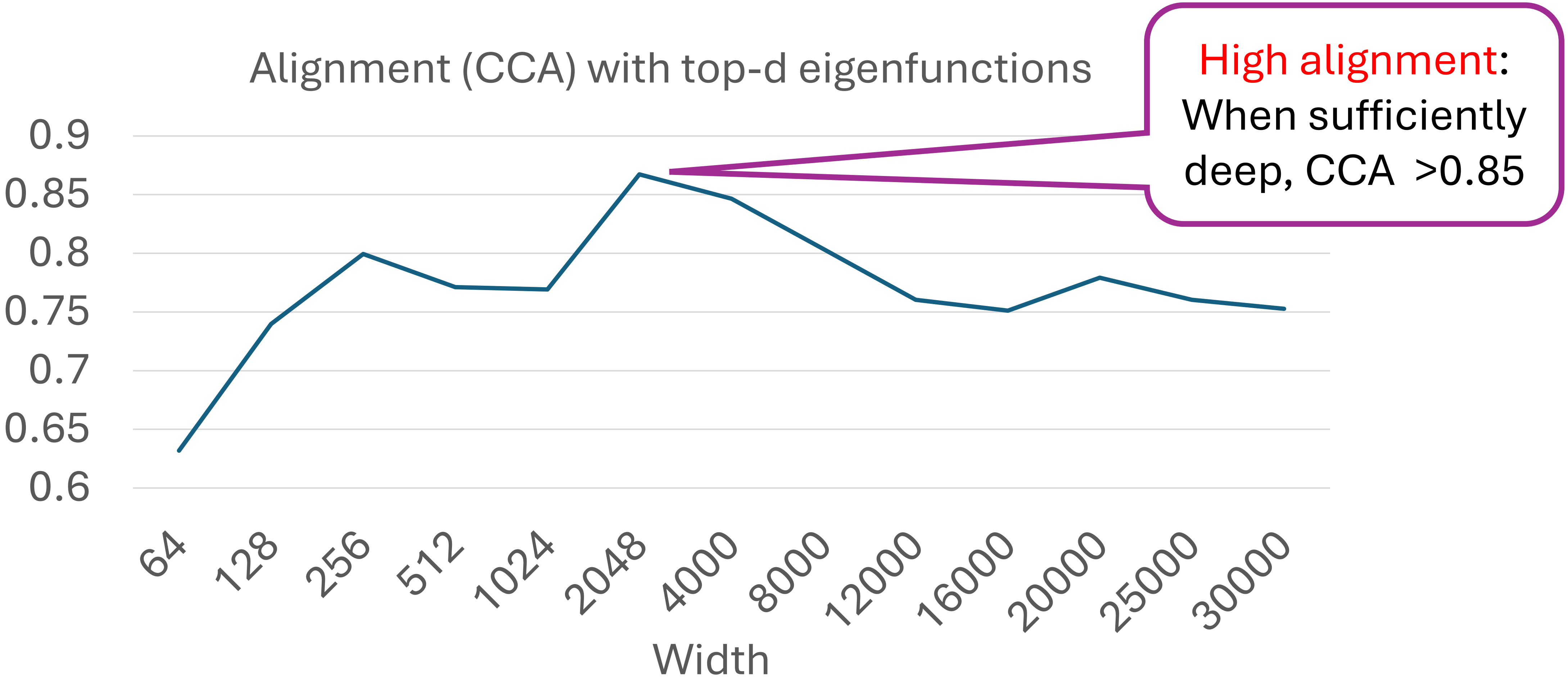
Experiment

- **Empirically verify that:**
  - High alignment: $span(\Phi) \approx span$(top-d eigenfunctions)
  - Diminishing return: Making a large model larger is useless
- Alignment metric: Canonical Correlation Analysis (CCA)
- Compare between:
  - Exact top-d eigenfunctions obtained by kernel PCA
  - The representation of a deep encoder Φ

Setting

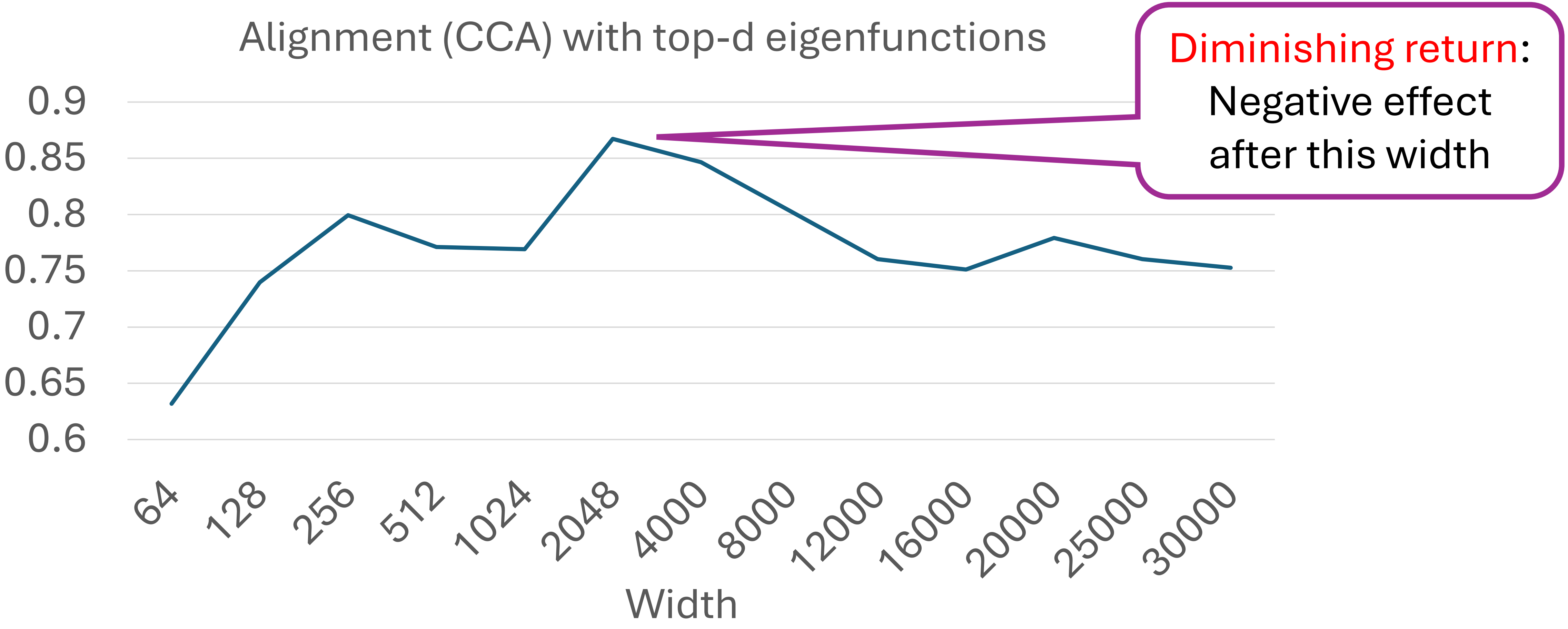
- Dataset: Abalone – a tabular dataset from OpenML
- Model: MLP
- Embedding dimension: d = 128
- Context: KNN with K = 30
- Pretraining: Non-contrastive learning

3-layer MLP, different widths

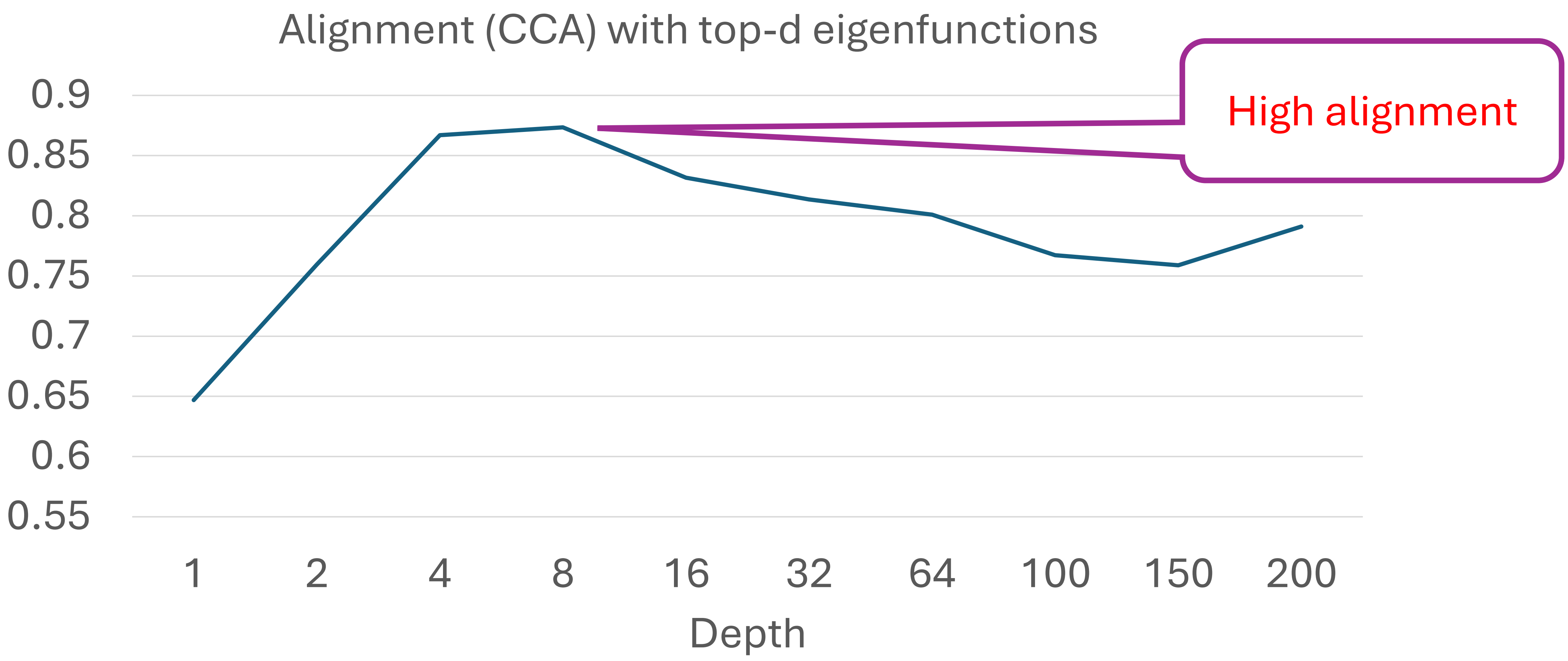


Carnegie Mellon University

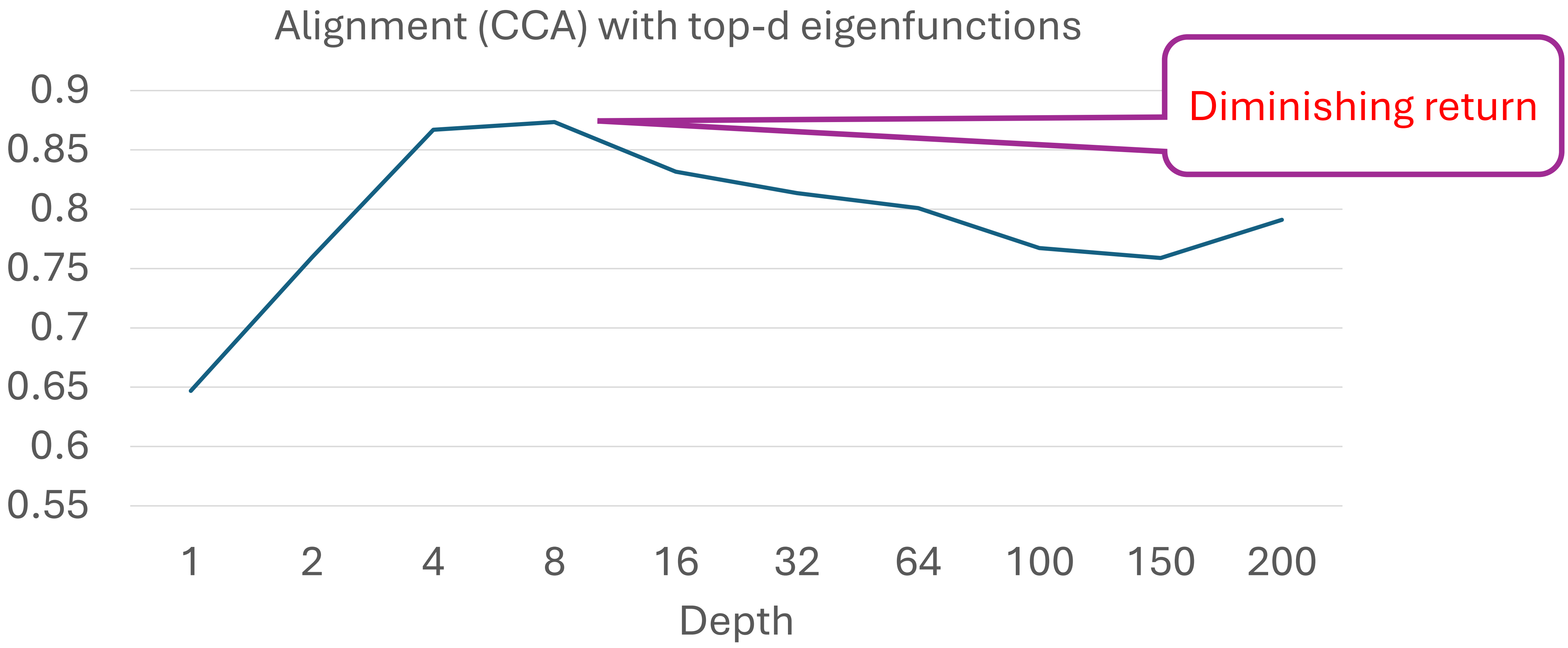
3-layer MLP, different widths



Width = 512, different depths



Width = 512, different depths



Conclusion

- When the model is large enough, further scaling has little use
- Further improvement requires better contexts

Part 4:
Towards better contexts

First question:

Which contexts are “better”?

Bad Contexts

- **Example 1:** A is independent of X
- **Example 2:** $A = X$
- Both contexts are clearly useless
- Key: The association strength between X and A
  - The context is useless if the association is too weak/strong

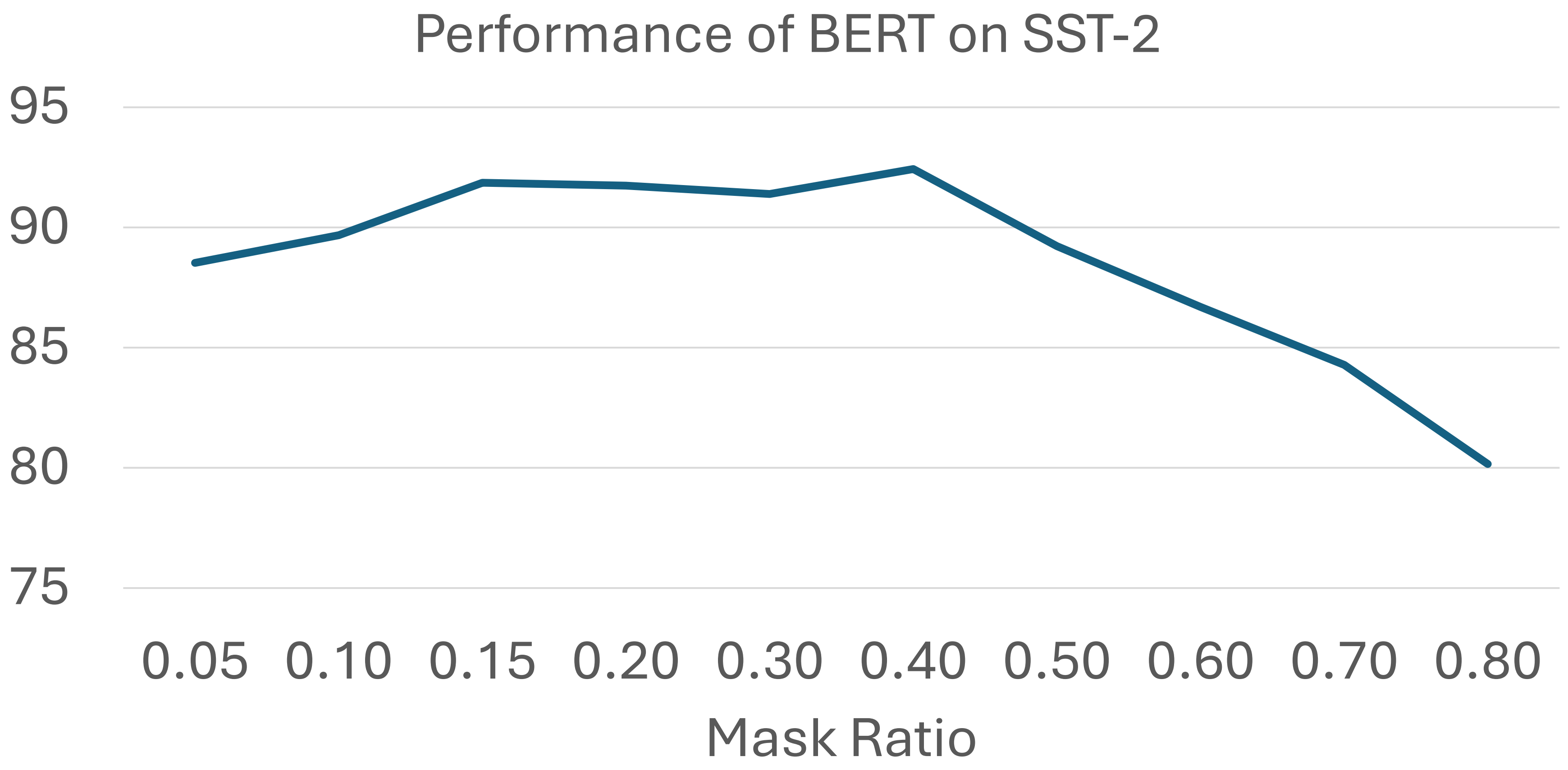
Result #6:

A good context must have a moderate association between X and A

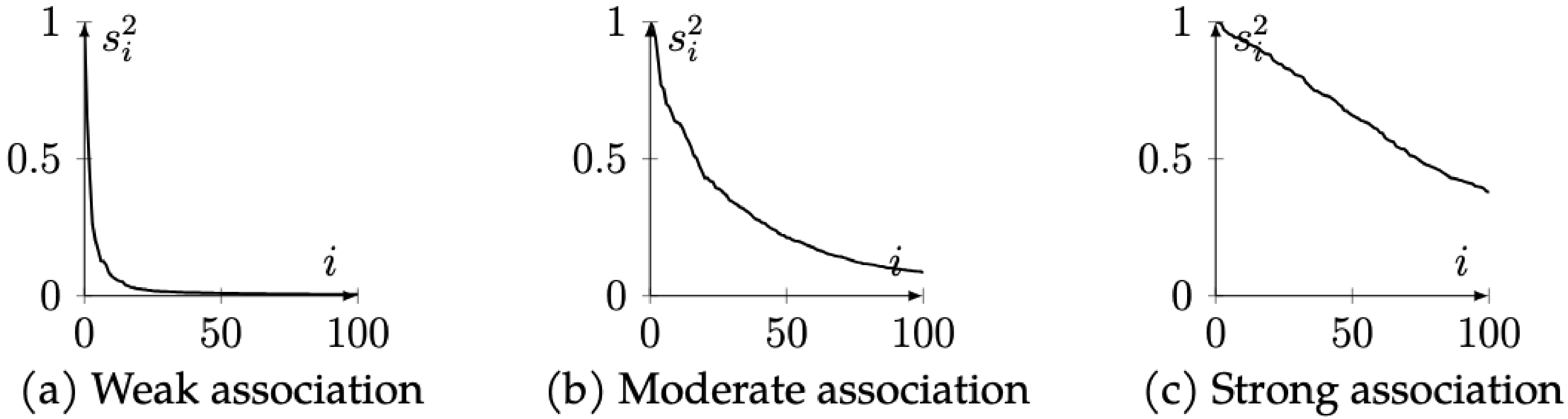
Carnegie Mellon University

Empirical Evidence

- BERT is the best when the mask ratio is neither too high nor too low



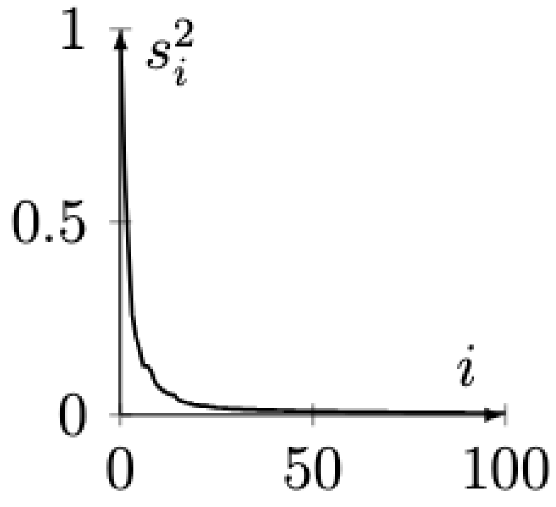
The association strength between X and A affects the decay rate of the eigenvalues



Carnegie Mellon University

Weak Association

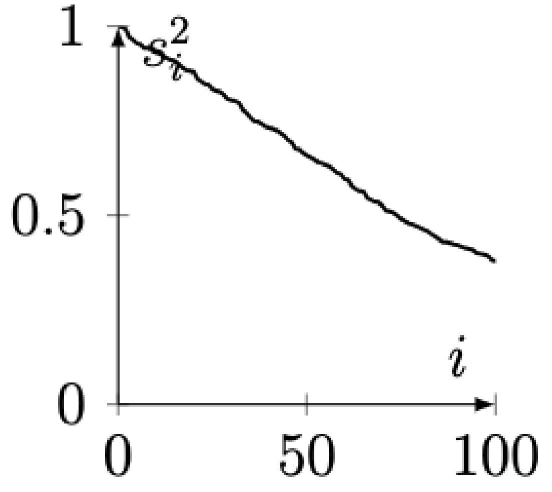
- Fast eigenvalue decay
- Very few tasks are compatible
- The encoder is not transferable



(a) Weak association

Strong Association

- Slow eigenvalue decay
- Need a larger embedding dimension d
- Leads to a higher sample complexity



(c) Strong association

How to obtain moderate association

- Suppose we have k contexts $P_1^+, \cdots P_k^+$
- But each context is too weak/strong, so no one is good
- Idea: Mix them together!

Carnegie Mellon University

Result #7:
Mixing multiple existing contexts
can give us better contexts

Three base operations of mixing contexts

- Concatenation
- Convolution
- Convex combination

Concatenation

- Pretrain individual encoders $\Phi_1, \Phi_2, \cdots, \Phi_k$ on k contexts
- $\Phi(x) = [\Phi_1(x), \Phi_2(x), \cdots, \Phi_k(x)]$
- Makes the association stronger

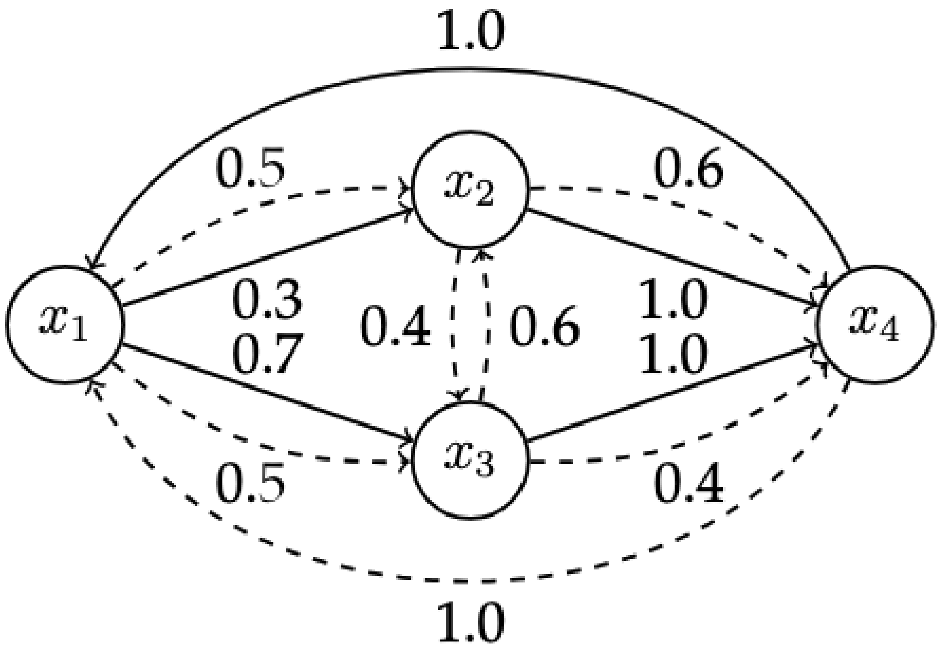
Convolution

- Analogous to composing data augmentations
  - Apply translation, rotation and cropping to the same image
- Makes the association weaker

Convex Combination

- Training objective of each individual context $\mathcal{R}_1, \cdots, \mathcal{R}_k$
- Convex combination: $\mathcal{R} = w_1\mathcal{R}_1 + \cdots + w_k\mathcal{R}_k$
- Balance weak and strong associations

Convolution & Convex Combination in Random Walk Contexts



- Two random walk contexts: Solid and dashed edges
- **Convolution** = Two-step walk: First solid, second dashed
- **Convex combination** = One-step walk: Solid with probability w_1

When to use each mixing operation?

Mixing operation	**When to use?**
Concatenation	All contexts are weak
Convolution	All contexts are strong
Convex combination	Mixed weak and strong

Closing Remarks

Carnegie Mellon University

Open Problems

- Inductive bias of model architecture
- Implicit bias of optimization
- Create completely new contexts from data, not heuristics
- Extend the theory to system 2 thinking (reasoning)

Conclusion: The contexture theory clarifies the mechanism of representation learning

- **Result #1:** Representations are learned from association
- **Result #2:** Contexture preserves the most information of T_{P^+}
- **Result #3:** The "deep learning" way of learning the contexture
- **Result #4:** Contexture is optimal for compatible tasks
- **Result #5:** Making models larger has diminishing returns
- **Result #6:** A good context should have a moderate association
- **Result #7:** Mixing contexts produces better contexts

Thank you

Carnegie Mellon University