中国人民大學
RENMIN UNIVERSITY OF CHINA

高瓴人工智能学院
Gaoling School of Artificial Intelligence

Towards Understanding How Transformers Learn In-context Through a Representation Learning Lens

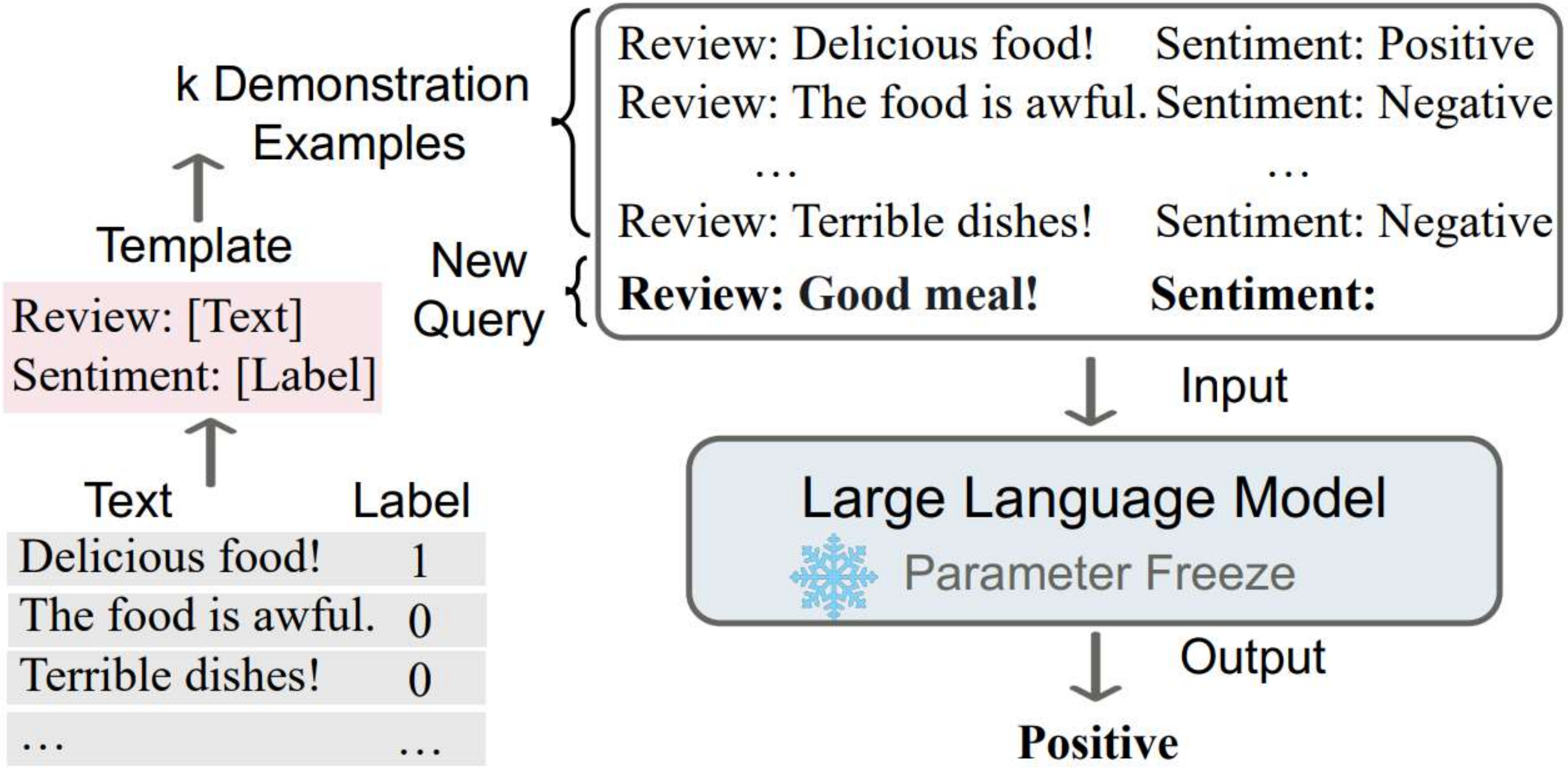
Ruifeng Ren and Yong Liu

Gaoling School of Artificial Intelligence,

Renmin University of China

Introduction

What's In-context learning (ICL)?



Dong Q, Li L, Dai D, et al. A survey for in-context learning[J]. arXiv preprint arXiv:2301.00234, 2022.

Introduction

One intuition is to think of it as an implicit gradient update.

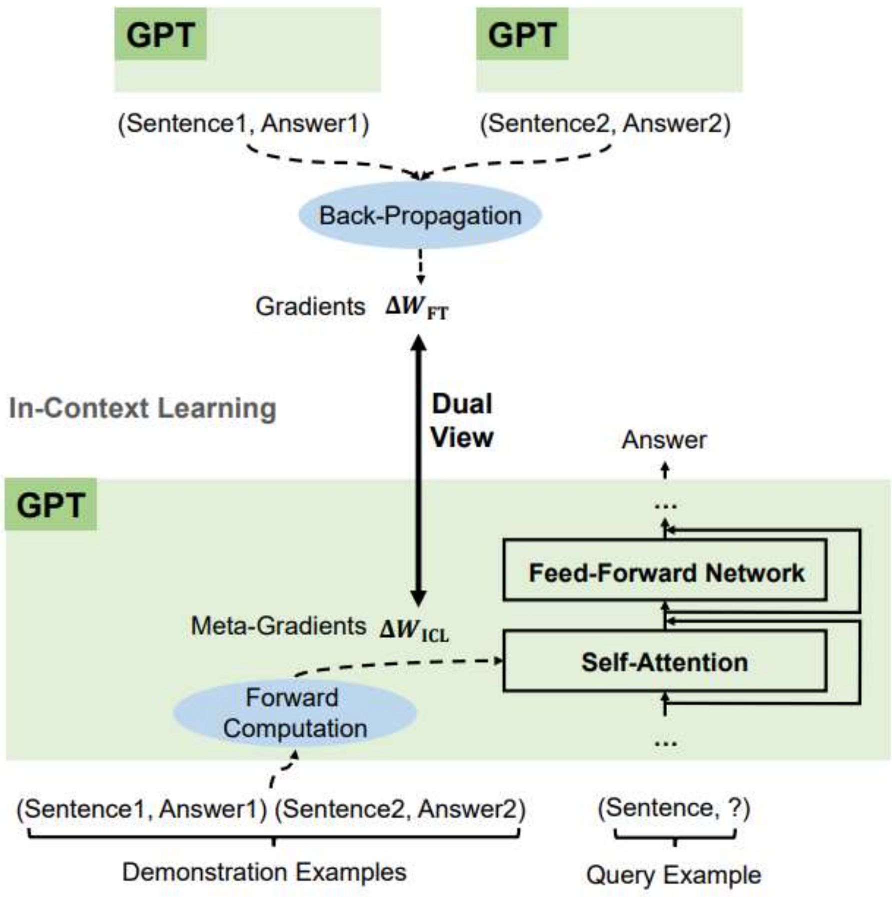
Fine-tuning: explicit gradient update

$$\widetilde{\mathcal{F}}_{\mathrm{FT}}(\mathbf{q}) = (W_V + \Delta W_V) X X^T (W_K + \Delta W_K)^T \mathbf{q}$$
$$= (W_{\mathrm{ZSL}} + \Delta W_{\mathrm{FT}})\, \mathbf{q},$$

Dual view

ICL: implicit gradient update

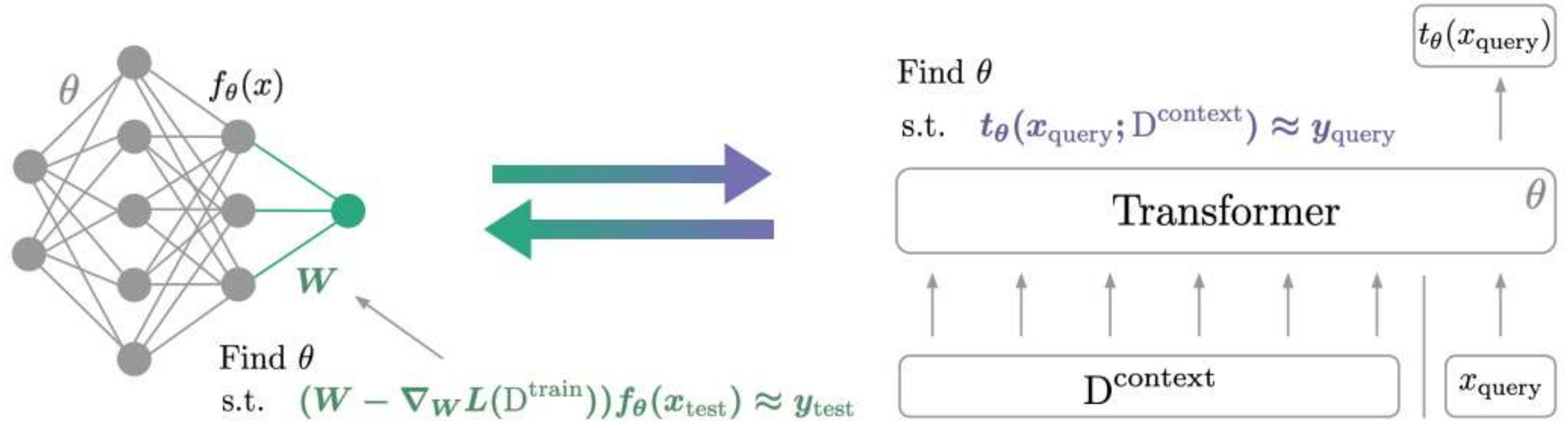
$$\widetilde{\mathcal{F}}_{\mathrm{ICL}}(\mathbf{q}) = W_{\mathrm{ZSL}}\mathbf{q} + W_V X' (W_K X')^T \mathbf{q}$$
$$= (W_{\mathrm{ZSL}} + \Delta W_{\mathrm{ICL}})\, \mathbf{q}.$$



Dai D, Sun Y, Dong L, et al. Why can gpt learn in-context? language models secretly perform gradient descent as meta optimizers[J]. arXiv preprint arXiv:2212.10559, 2022.

Introduction

One intuition is to think of it as an implicit gradient update.



$$e_j \leftarrow e_j + \mathrm{LSA}_\theta(\{e_1, \ldots, e_N\}) = e_j + \sum_h \underline{P_h V_h K_h^T} q_{h,j} = e_j + \sum_h P_h \sum_i \underline{v_{h,i} \otimes k_{h,i}} q_{h,j}$$

Linear attention setting

$$\begin{pmatrix} x_j \\ y_j \end{pmatrix} \leftarrow \begin{pmatrix} x_j \\ y_j \end{pmatrix} + \frac{\eta}{N} I \sum_{i=1}^{N} \left(\begin{pmatrix} 0 & 0 \\ W_0 & -I_y \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} \right) \otimes \left(\begin{pmatrix} I_x & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix} \right) \begin{pmatrix} I_x & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_j \\ y_j \end{pmatrix}$$

$$= \begin{pmatrix} x_j \\ y_j \end{pmatrix} + \frac{\eta}{N} I \sum_{i=1}^{N} \begin{pmatrix} 0 \\ W_0 x_i - y_i \end{pmatrix} \otimes \begin{pmatrix} x_i \\ 0 \end{pmatrix} \begin{pmatrix} x_j \\ 0 \end{pmatrix} = \begin{pmatrix} x_j \\ y_j \end{pmatrix} + \begin{pmatrix} 0 \\ -\Delta W x_j \end{pmatrix}.$$

Specific weight construction

Von Oswald J, Niklasson E, Randazzo E, et al. Transformers learn in-context by gradient descent[C]//International Conference on Machine Learning. PMLR, 2023: 35151-35174.

Introduction

The drawbacks of existing methods:

(i) Interpret ICL as implicit fine-tuning:

- This comparison is a formal resemblance and specific details are ambiguous;
- ICL is unsupervised, whereas fine-tuning is a supervised process;

(ii) The ability to implement the gradient descent algorithm:

- Specific tasks (linear regression), specific weight constructions

Can we relate ICL to gradient descent:

- under the softmax attention setting, rather than the linear attention setting
- without assuming specific constructions for specific tasks

Connecting Softmax Attention with Kernels

For the query input $\boldsymbol{x}'_{T+1}$, the output of one attention layer is

$$\boldsymbol{h}'_{T+1} = \boldsymbol{W}_V \boldsymbol{X} \boldsymbol{A} = \boldsymbol{W}_V \boldsymbol{X} \operatorname{softmax}\left(\frac{(\boldsymbol{W}_K \boldsymbol{X})^T \boldsymbol{W}_Q \boldsymbol{x}'_{T+1}}{\sqrt{d_o}}\right)$$

The attention part can be viewed as the product of two parts

$$\boldsymbol{A} = \boldsymbol{A}_u \boldsymbol{D}^{-1}, \quad \boldsymbol{A}_u = \exp\left((\boldsymbol{W}_K \boldsymbol{X})^T \boldsymbol{W}_Q \boldsymbol{X} / \sqrt{d_o}\right), \quad \boldsymbol{D} = \operatorname{diag}(\boldsymbol{1}_N^T \boldsymbol{A}_u)$$

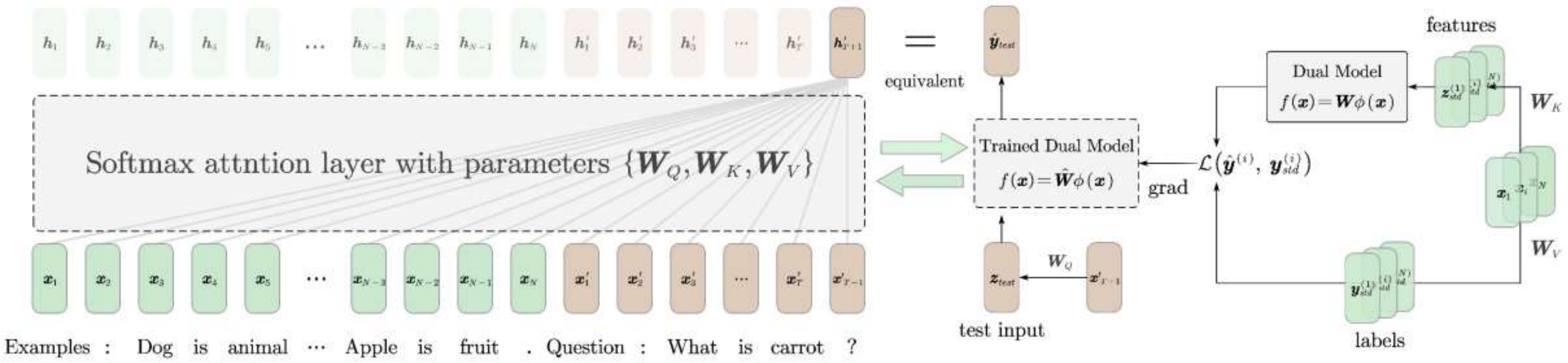
Each entry can be seen as the output of kernel K_{sm} defined for the mapping ϕ

$$\boldsymbol{A}_u(i,j) = \exp\left((\boldsymbol{W}_K \boldsymbol{x}_i)^T \boldsymbol{W}_Q \boldsymbol{x}_j\right) = K_{sm}(\boldsymbol{W}_K \boldsymbol{x}_i,\ \boldsymbol{W}_V \boldsymbol{x}_j) = \phi(\boldsymbol{W}_K \boldsymbol{x}_i)^T \phi(\boldsymbol{W}_Q \boldsymbol{x}_j)$$

Furthermore , we can use this mapping to construct the dual model as

$$f(\boldsymbol{x}) = \boldsymbol{W} \phi(\boldsymbol{x})$$

The Gradient Descent Process of ICL

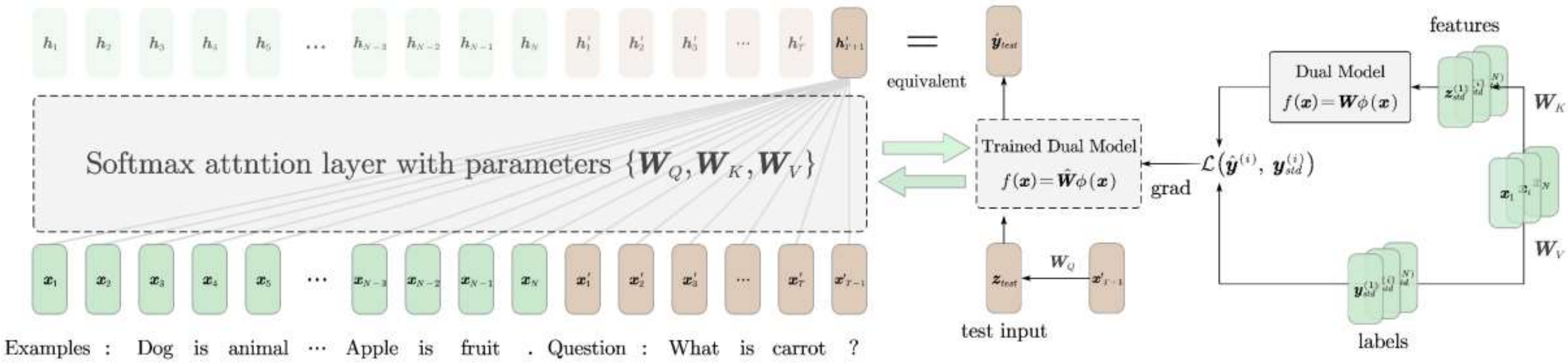


Theorem 3.1. *The query token h'_{T+1} obtained through ICL inference process with one softmax attention layer, is equivalent to the test prediction $\hat{y}_{test}$ obtained by performing one step of gradient descent on the dual model $f(x) = W\phi(x)$. The form of the loss function $\mathcal{L}$ is:*

$$\mathcal{L} = -\frac{1}{\eta D}\sum_{i=1}^{N}(W_V x_i)^T W\phi(W_K x_i), \tag{9}$$

where η is the learning rate and D is a constant.

The Gradient Descent Process of ICL



Forward perspective:

$$h'_{T+1} = W_V X \, \text{softmax}\left(\frac{(W_K X)^T W_Q x'_{T+1}}{\sqrt{d_o}}\right)$$

$$\exp(x^T y) = K_{\exp}(x, y) = [\phi(x)]^T \phi(y)$$

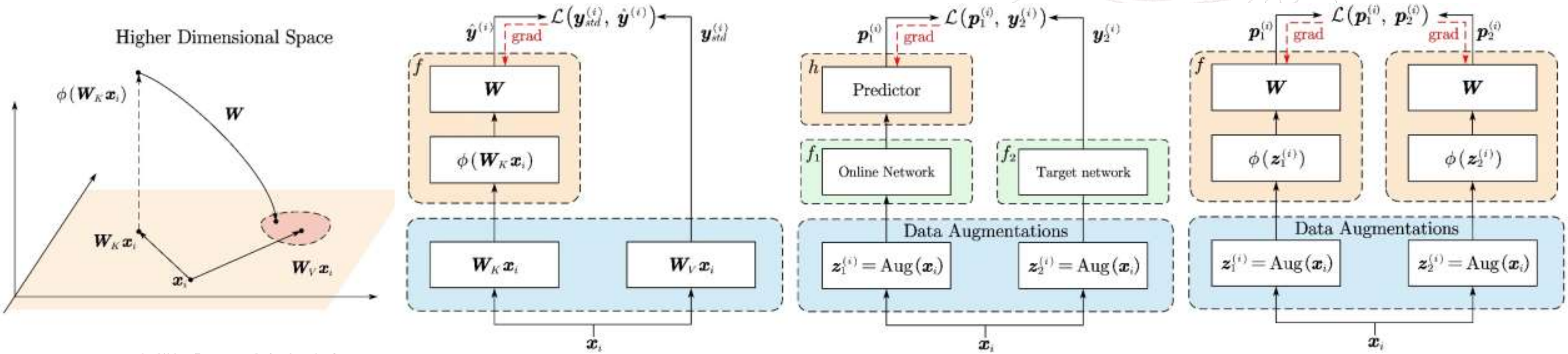
Backward perspective:

Dual model: $f(x) = W\phi(x)$

Loss: $\mathcal{L} = -\dfrac{1}{\eta D}\sum_{i=1}^{N}(W_V x_i)^T W\phi(W_K x_i)$

Test output: $\hat{y}_{test} = \hat{f}(W_Q x'_{T+1}) = \widehat{W}\phi(W_Q x'_{T+1})$

The Gradient Descent Process of ICL



Left Part: The representation learning process for the ICL inference by one attention layer. **Remaining Part:** Comparison of the ICL Representation Learning Process **(Center Left)**, Contrastive Learning without Negative Samples **(Center Right)**, and Contrastive Kernel Learning **(Right).**

Chen X, He K. Exploring simple siamese representation learning[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 15750-15758.
Esser P, Fleissner M, Ghoshdastidar D. Non-Parametric Representation Learning with Kernels[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(11): 11910-11918.

Generalization Bound

Gaoling School of Artificial Intelligence

Generalization Bound of the dual gradient descent process for ICL

Generally, we consider the representation learning loss as

$$\mathcal{L}(f) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\mathcal{T}}}\left[-\left(\boldsymbol{W}_V\boldsymbol{x}\right)^T f(\boldsymbol{x})\right] = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\mathcal{T}}}\left[-\left(\boldsymbol{W}_V\boldsymbol{x}\right)^T \boldsymbol{W}\phi(\boldsymbol{W}_K\boldsymbol{x})\right], \tag{10}$$

where $f \in \mathcal{F}$ and $\mathcal{D}_{\mathcal{T}}$ is the distribution for some ICL task $\mathcal{T}$.

Theorem 3.2. *Define the function class as* $\mathcal{F} := \{f(\boldsymbol{x}) = \boldsymbol{W}\phi(\boldsymbol{W}_K\boldsymbol{x}) \mid \|\boldsymbol{W}\| \le w\}$ *and let the loss function defined as Eq (10). Consider the given demonstration set as* $\mathcal{S} = \{\boldsymbol{x}_i\}_{i=1}^N$ *where* $\mathcal{S} \subseteq \mathcal{S}_{\mathcal{T}}$ *and* $\mathcal{S}_{\mathcal{T}}$ *is all possible demonstration tokens for some task* $\mathcal{T}$*. With the assumption that* $\|\boldsymbol{W}_V\boldsymbol{x}_i\|, \|\boldsymbol{W}\phi(\boldsymbol{W}_K\boldsymbol{x}_i)\| \le \rho$*, then for any* $\delta > 0$*, the following statement holds with probability at least* $1-\delta$ *for any* $f \in \mathcal{F}$

$$\mathcal{L}(\hat{f}) \le \mathcal{L}(f) + O\left(\frac{w\rho d_o\sqrt{\mathrm{Tr}(\boldsymbol{K}_{\mathcal{S}})}}{N} + \sqrt{\frac{log\frac{1}{\delta}}{N}}\right). \tag{11}$$

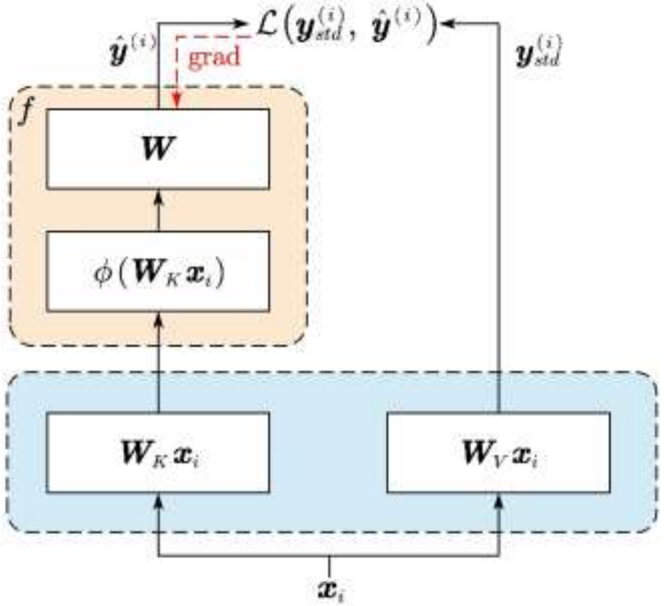
Attention Modification

Attention Modification Inspired by the Representation Learning Lens

Original: $\mathcal{L} = -\frac{1}{\eta D}\sum_{i=1}^{N}(\boldsymbol{W}_V\boldsymbol{x}_i)^T\boldsymbol{W}\phi(\boldsymbol{W}_K\boldsymbol{x}_i)$

Modified: $\mathcal{L} = -\frac{1}{\eta D}\sum_{i=1}^{N}[g_1(\boldsymbol{W}_V\boldsymbol{x}_i)]^T\boldsymbol{W}\phi(g_2[\boldsymbol{W}_K\boldsymbol{x}_i])$

Modified model: $\boldsymbol{h}'_{T+1} = g_1(\boldsymbol{W}_V\boldsymbol{X})\mathrm{softmax}\left(\frac{[g_2(\boldsymbol{W}_K\boldsymbol{X})]^T\boldsymbol{W}_Q\boldsymbol{x}'_{T+1}}{\sqrt{d_o}}\right)$

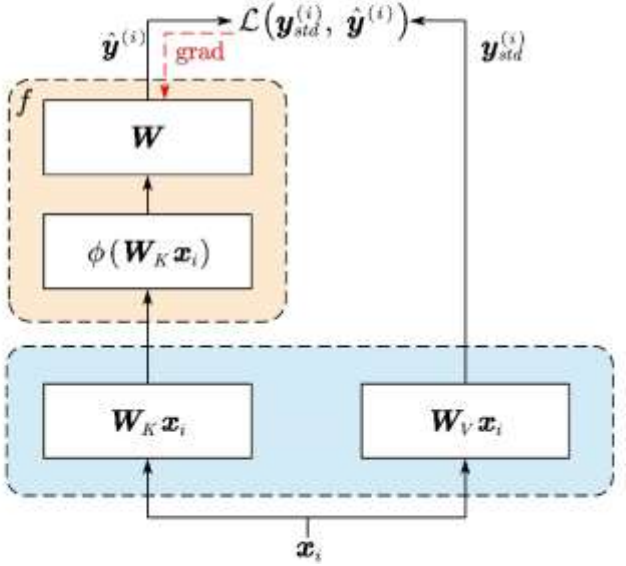


For example, we can take $g(\boldsymbol{W}\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x} + c\boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{x})$ **(Parallel Adapter)**

For different tasks, the augmentation approach should be specifically designed to adapt them.

He J, Zhou C, Ma X, et al. Towards a unified view of parameter-efficient transfer learning. ICLR 2022

Attention Modification Inspired by the Representation Learning Lens



Modified: $\mathcal{L} = -\frac{1}{\eta D} \sum_{i=1}^{N} (\boldsymbol{W}_V \boldsymbol{x}_i)^T \boldsymbol{W} \phi(\boldsymbol{W}_K \boldsymbol{x}_i) + \alpha \|\boldsymbol{W}\|_F^2$

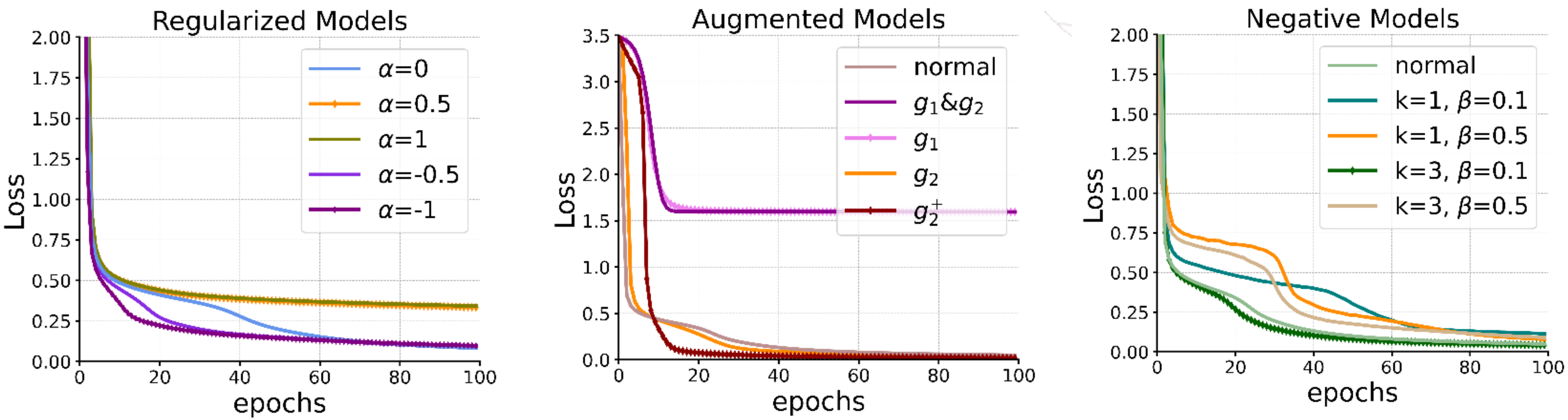
Regularized model: $\boldsymbol{h}'_{T+1} = \boldsymbol{W}_V \boldsymbol{X} \left[\text{softmax}\left(\frac{(\boldsymbol{W}_K \boldsymbol{X})^T \boldsymbol{W}_Q \boldsymbol{x}'_{T+1}}{\sqrt{d_o}} \right) - \alpha \boldsymbol{I} \right]$

Modified: $\mathcal{L} = -\frac{1}{\eta D} \sum_{i=1}^{N} \left[(\boldsymbol{W}_V \boldsymbol{x}_i)^T \boldsymbol{W} \phi(\boldsymbol{W}_K \boldsymbol{x}_i) - \frac{\beta}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} (\boldsymbol{W}_V \boldsymbol{x}_j)^T \boldsymbol{W} \phi(\boldsymbol{W}_K \boldsymbol{x}_j) \right]$

Negative model: $\boldsymbol{h}'_{T+1} = \boldsymbol{W}_V \tilde{\boldsymbol{X}} \left[\text{softmax}\left(\frac{(\boldsymbol{W}_K \boldsymbol{X})^T \boldsymbol{W}_Q \boldsymbol{x}'_{T+1}}{\sqrt{d_o}} \right) - \alpha \boldsymbol{I} \right]$, where $\tilde{\boldsymbol{X}}^{(i)} = \tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i - \frac{\beta}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \boldsymbol{x}_j$

Experiment results

高瓴人工智能学院
Gaoling School of Artificial Intelligence



The performance for regularized models (Left), augmented models (Center) and negative models (Right) with different settings.

THANK YOU!