In-Context Learning behaves as a greedy layer-wise gradient descent algorithm

Brian K Chen, Tianyang Hu, Hui Jin, Hwee Kuan Lee, Kenji Kawaguchi

NUS National University of Singapore

Bioinformatics Institute BII

HUAWEI

Motivation

- ICL is an emergent property found in LLMs
- Appending prompts to existing ones "teaches" the LLM new information without training the model

Circulation revenue has increased by 5% in Finland. // Positive	Circulation revenue has increased by 5% in Finland. // Finance
Panostaja did not disclose the purchase price. // Neutral	They defeated … in the NFC Championship Game. // Sports
Paying off the national debt will be extremely painful. // Negative	Apple … development of in-house chips. // Tech
The company anticipated its operating profit to improve. // ________	The company anticipated its operating profit to improve. // ________
LM	LM
Positive	Finance

- Theoretical understanding of ICL remains limited
- Existing work that considers ICL as a single step of gradient descent focuses on limited cases with specific weights
- Want to study the mechanics of ICL by looking at the linearized attention module with generic weights

Preliminaries

Linearization of the attention mechanism:

$$LinAttn(V, \phi(K), \phi(Q)) = V\phi(K)^T\phi(Q)$$

Where $\phi(x)^T\phi(y)$ is a kernel approximator and the scaling factor is omitted

- Linearized Attention greatly reduces computational cost
- There has been a lot of work in this field which has shown promise in recent years
- E.g. Retnet, Infini-Attention, Hedgehog etc.

Dual form of linear attention and gradient decent:

Let $f_W(x) = Wx$ be a linear function. Given gradient descent with l_2 loss, T training samples $\{x_i, y_i\}_{i=1}^T$ and learning rate η

$$W_1 x = \left(W_0 - \eta\nabla \frac{1}{T}\sum_{i=1}^{T} l_2(f_W(x_i), y_i)\Big|_{W=W_0} \right) x$$
$$= W_0 x + LinAttn(\frac{\eta}{T}E, X, x)$$

$X = [x_1; \dots; x_T]$ is the matrix of inputs

$E = Y - W_0 X$ is the error matrix where $Y = [y_1, \dots, y_T]$

Main Result

Theorem 1:

For an initial self-attention mechanism with query token $\boldsymbol{q}$, matrices W_V, W_K, prompt $X = [p_N, \dots, p_1]$ and operator $\mathcal{F}_0([X], \boldsymbol{q}) = LinAttn(W_V X, \phi(W_K X), \boldsymbol{q})$, the following systems are equivalent:

$$S_0 = \mathcal{F}_0([X'; X], \boldsymbol{q})$$

and

$$S_1 = \mathcal{F}_1([X], \boldsymbol{q})$$

Where $\mathcal{F}_1([X], .)$ is the linear function $\mathcal{F}_0([X], .)$ after one step of gradient descent with learning rate η and training set $\{x_i, y_i\}_{i=1}^M$.
For every $i \in \{1, \dots, M\}$:

$$x_i = \phi(W_K p_i')$$
$$y_i = \frac{M}{\eta} W_V p_i' + \mathcal{F}_0([X], \phi(W_K p_i'))$$

Observations:

- In-context learning forms a type of meta-optimizer on the query resembling gradient descent with specific training data for linearized transformers
- Statement isn't constrained to specific regression settings and values for W_Q, W_K, W_V
- $W_V p_i'$ is intuitively the "value" which we place upon token $W_Q p_i'$. Here we place emphasis on $W_K p_i'$ rather than the query token itself.

Extension to Multiple Layers:

- Consider a more realistic model architecture with L layers stacked upon each other
- $f(x) = (T_L + I) \circ \dots \circ (T_1 + I)(x)$
- For all i Layer T_i is either a FFN layer or a linear self-attention layer $T_i = LSA_i$
- Corresponding weight matrices $W_{Q_i}, W_{K_i}, W_{V_i}$
- I is the identity function

Algorithm 1: ICL imitation algorithm

input: f_1 and $[p_m', \dots, p_1', p_n, \dots, p_1]$
for $i \in \{1, \dots, L\}$
IF T_i is a FFN with residual connection, return
$[p_m', \dots, p_1', p_n, \dots, p_1] = (T_i + I)([p_m', \dots, p_1', p_n, \dots, p_1])$

ELSE $T_i = LSA_i$
(a) construct matrix $W_0 = W_{base,i}([p_n, \dots, p_1])$
(b) Update the linear functional $f(x) = W_0 x$ with a single step of gradient descent with learning rate m and training set $\{\phi(), W_{V_i} p_j' + W_0\phi(W_{K_i} p_j'))\}_{j=1}^m$ such that the updated weights are W_1
(c) $[p_m', \dots, p_1', p_n, \dots, p_1] = W_1\phi(W_{Q_i}[p_m', \dots, p_1', p_n, \dots, p_1]) + [p_m', \dots, p_1', p_n, \dots, p_1]$

Theorem 2:

For a model f_1 described above and a prompt $[p_m', \dots, p_1', p_n, \dots, p_1]$, Algorithm 1 produces the same output as $f_1(p_m', \dots, p_1', p_n, \dots, p_1)$

Connection to greedy layer-wise algorithms

- Algorithm 1 is a recursive algorithm The labels are generated from the inputs themselves
- Algorithm 1 takes the form of a greedy unsupervised layer-wise pretraining algorithm (GLT)
- Real life phenomena shows degree of similarity:
  - GLT are observed to achieve quick convergence, ICL only applies a single step of gradient descent
  - GLT shown to help learn internal representations that represent higher level abstractions, ICL has similar properties
- This may motivate future steps forward with ICL:
  - This suggests that ICL may be a form of initialization which projects the model into a specific space
  - May want to have fixed ICL terms to guide fine-tuning steps, serving as the initialization. This actually begins to resemble instruction tuning
  - Motivates the construction of a unified theoretical framework combining fine-tuning methods with ICL and instruction tuning

Katharopoulos, A.,et al. Transformers are rnns: Fast autoregressive transformers with linar eattention. In International conference on machine learning, pp. 5156–5165. PMLR, 2020.
Sam L., Kexin J. 2022. Lecture 07: [Towards Understanding In-context Learning]. Accessed July 11, 2024.
Kazuki Irie, Róbert Csordás, & Jürgen Schmidhuber. (2022). The Dual Form of Neural Networks Revisited: Connecting Test Time Predictions to Training Patterns via Spotlights of Attention.