LLM as a function approximator for $f(x;\theta)$ where θ is in *natural language* ?!

Verbalized Machine Learning: Revisiting Machine Learning with Language Models

Tim Z. Xiao[1,2] Robert Bamler[2] Bernhard Schölkopf[1] Weiyang Liu[1,3]
[1]Max Planck Institute for Intelligent Systems, Tübingen [2]University of Tübingen [3]University of Cambridge

Idea Summary

- LLM as a function approximator
- Parameter Space: *Natural Language*
- Verbalized Inference – $f_{\text{model}}(\,\cdot\,;\boldsymbol{\theta})$
- Verbalized Optimization – $f_{\text{opt}}(\,\cdot\,;\boldsymbol{\psi})$

Algorithm 1 Training in VML

Initialize model parameters $\boldsymbol{\theta}_0$, iteration number T, batch size M and optimizer parameters $\boldsymbol{\psi}$;
for $i = 1, \cdots, T$ **do**
  Sample M training examples $x_1, \cdots, x_M$;
  for $m = 1, 2, \cdots, M$ **do**
    $\hat{y}_m = f_{\text{model}}(x_m; \boldsymbol{\theta}_{i-1})$;
  end
  $\boldsymbol{\theta}_i = f_{\text{opt}}(\{x_m, \hat{y}_m, y_m\}_{m=1}^{M}, \boldsymbol{\theta}_{i-1}; \boldsymbol{\psi})$;
end

Text prompt template for the learner

You are the model. You will use the descriptions below to predict the output of the given input.

** Pattern Descriptions: **
You are designed to do regression, i.e., to predict the output of any given input. Both input and output are real numbers. — Model parameters θ

** Input: **
[0.59] — Data x

Please give your output strictly in the following format:

```
Explanations: [Your step-by-step analyses and results]

Output:
[Your output MUST be in REAL NUMBER ROUNDED TO TWO DECIMAL POINTS; make necessary assumptions if needed; it MUST be in the same format as the Input]
```

Please ONLY reply according to this format, don't give me any other words.

Text prompt template for the optimizer

You are the optimizer for a model, your goal is to learn the best descriptions for the model. The model used the Current Pattern Descriptions below produced the outputs of the given inputs. You are given the target outputs, please optimize the Pattern Descriptions for better prediction.

** Inputs (a batch of i.i.d. data): **
[[0.59] [1.55] [0.64] [1.43] [0.28] [0.02] [0.84] [0.39] [0.02] [1.28]] — A batch of data $\{x_1, \ldots, x_n\}$

** Current Pattern Descriptions: **
You are designed to do regression, i.e., to predict the output of any given input. Both input and output are real numbers. — Model parameters θ

** The model outputs: **
[[0.59] [3.88] [1.28] [1.43] [0.53] [0.02] [1.] [0.39] [1.] [0.]] — Prediction $\{\hat{y}_1, \ldots, \hat{y}_n\}$

** The target outputs: **
[[5.84] [8.51] [5.92] [8.09] [4.98] [3.91] [6.46] [5.23] [3.88] [7.88]] — Ground Truth $\{y_1, \ldots, y_n\}$

If the model is doing well, you can keep using the current descriptions. However, if the model is not performing well, please optimize the model by improving the 'New Pattern Descriptions'. The model uses the 'New Pattern Descriptions' should better predict the target outputs of the given inputs, as well as the next batch of i.i.d. input data from the same distribution. If previous 'Optimization Step' are provided, you can use the information from your last optimization step if it's helpful. Please think step by step and give your outputs strictly in the following format: — Optimizer parameter ψ

```
Reasoning:
[be explicit and verbose, improve the Current Pattern Descriptions by yourself;]

New Pattern Descriptions:
[put your new descriptions here; MUST be specific and concrete;]
```

Verbalized loss function

Please ONLY reply according to this format, don't give me any other words.

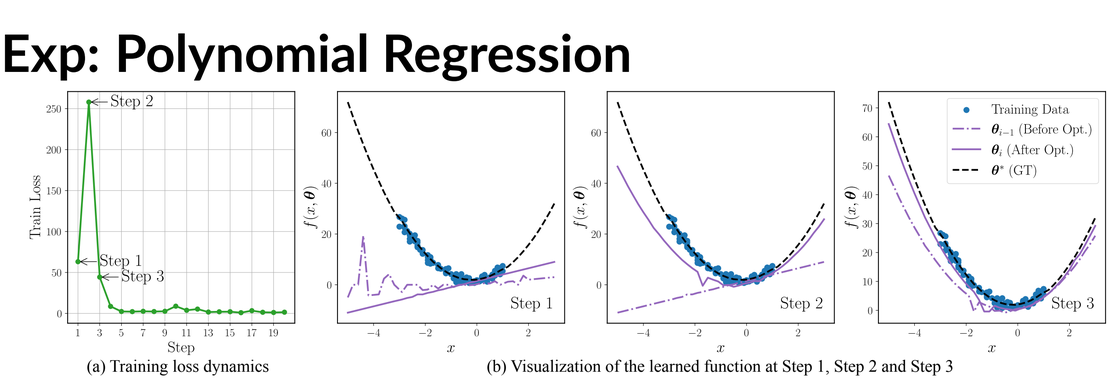
Learner LLM f_{model} — update model parameters — Optimizer LLM f_{opt} — provide inference results — **Iterative optimization**

Figure 1 : An example of iterative optimization and text prompt templates of the learner $f_{\text{model}}(\,\cdot\,;\boldsymbol{\theta})$ and the optimizer $f_{\text{opt}}(\,\cdot\,;\boldsymbol{\psi})$ in regression.

Advantages

1. *Easy encoding of inductive bias:* prior knowledge about the problem and hypothesis class can be encoded in natural language and fed into the LLM-parameterized learner;
2. *Automatic model class selection:* the optimizer can automatically select and update a concrete model class based on data and verbalized prior knowledge during training;
3. *Interpretable learner updates:* the optimizer provides explanations for each update.

Exp: Polynomial Regression



(a) Training loss dynamics (b) Visualization of the learned function at Step 1, Step 2 and Step 3

Model parameters θ_0 at Step 1
Model parameter initialization
You are designed to do regression, i.e., to predict the output of any given input. Both input and output are real numbers.

Optimizer output at Step 1
Reasoning:
After examining the model outputs and target outputs, it's clear that the model is not performing well. The model outputs are mostly around 0, while the target outputs have a larger range of values. This suggests that the model is not capturing the underlying pattern in the data.

Upon closer inspection, I notice that the inputs and target outputs seem to have a positive correlation. This means that as the input increases, the target output also tends to increase.

To improve the model, I will introduce a new pattern description that captures this positive correlation. I will use a simple linear relationship between the input and output, with a slope and intercept that can be learned from the data.

New Pattern Descriptions: — Model parameters θ_1
The new pattern description is: output = 2.5 * input + 1.5.

Model parameters θ_1 at Step 2
The new pattern description is: output = 2.5 * input + 1.5.

Optimizer output at Step 2
Reasoning:
After examining the model outputs and target outputs, it's clear that the model is not performing well. The model outputs are not close to the target outputs, and the pattern description output = 2.5 * input + 1.5 seems to be oversimplifying the relationship between the input and output.

Upon closer inspection, I notice that the target outputs are generally larger than the model outputs. This suggests that the model is underestimating the target outputs. I also notice that the target outputs seem to have a non-linear relationship with the inputs, with larger inputs corresponding to larger outputs, but not in a linear way.

To improve the model, I will introduce a new pattern description that captures this non-linear relationship. I will use a quadratic relationship between the input and output, with coefficients that can be learned from the data. This will allow the model to better fit the curved relationship between the inputs and target outputs.

New Pattern Descriptions: — Model parameters θ_2
The new pattern description is: output = 2.2 * input^2 + 1.8 * input + 0.6.

Model parameters θ_2 at Step 3
The new pattern description is: output = 2.2 * input^2 + 1.8 * input + 0.6.

Optimizer output at Step 3
Reasoning:
After examining the model outputs and target outputs, it's clear that the model is not performing well. The model outputs are not close to the target outputs, and the pattern description output = 2.2 * input^2 + 1.8 * input + 0.6 seems to be undershooting the target outputs.

Upon closer inspection, I notice that the target outputs are generally larger than the model outputs, especially for larger inputs. This suggests that the model is underestimating the target outputs, especially for larger inputs. I also notice that the target outputs seem to have a stronger quadratic relationship with the inputs, with larger inputs corresponding to even larger outputs.

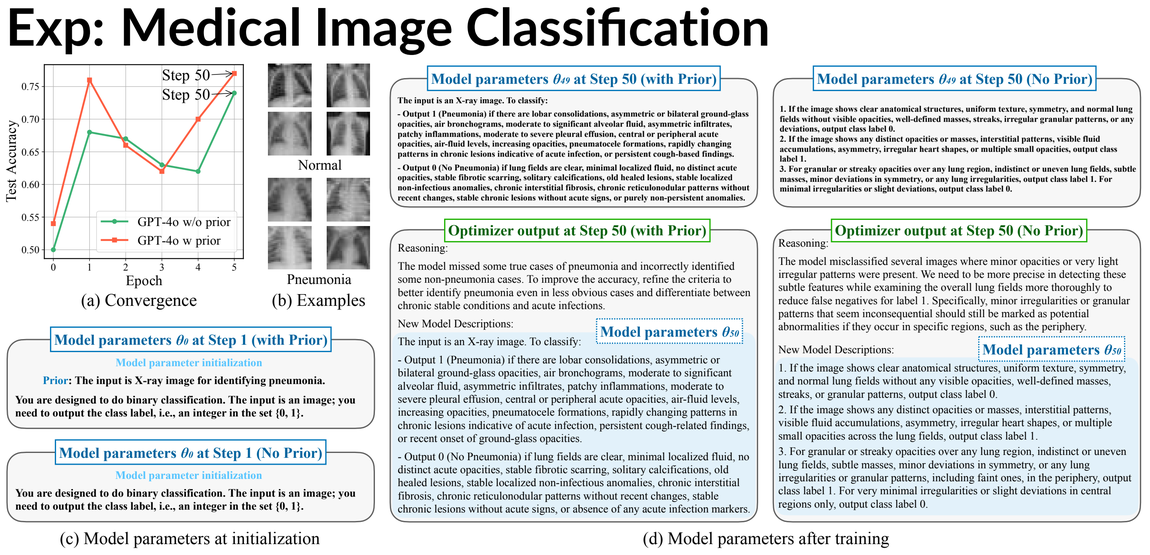
To improve the model, I will introduce a new pattern description that captures this stronger quadratic relationship. I will increase the coefficient of the quadratic term to capture the stronger curvature of the relationship. I will also adjust the linear term to better fit the data.

New Pattern Descriptions: — Model parameters θ_3
The new pattern description is: output = 2.8 * input^2 + 1.2 * input + 0.4.

(c) Model parameters and optimizer output at Step 1, Step 2 and Step 3

Figure 2 : Demo for polynomial regression and auto model class selection.

Exp: Medical Image Classification



(a) Convergence (b) Examples

Model parameters θ_0 at Step 1 (with Prior)
Model parameter initialization
Prior: The input is X-ray image for identifying pneumonia.
You are designed to do binary classification. The input is an image; you need to output the class label, i.e., an integer in the set {0, 1}.

Model parameters θ_0 at Step 1 (No Prior)
Model parameter initialization
You are designed to do binary classification. The input is an image; you need to output the class label, i.e., an integer in the set {0, 1}.

(c) Model parameters at initialization

Model parameters θ_{49} at Step 50 (with Prior)
The input is an X-ray image. To classify:
- Output 1 (Pneumonia) if there are lobar consolidations, asymmetric or bilateral ground-glass opacities, air bronchograms, moderate to significant alveolar fluid, asymmetric infiltrates, patchy inflammations, moderate to severe pleural effusion, central or peripheral acute opacities, air-fluid levels, increasing opacities, pneumatocele formations, rapidly changing patterns in chronic lesions indicative of acute infection, or persistent cough-based findings.
- Output 0 (No Pneumonia) if lung fields are clear, minimal localized fluid, no distinct acute opacities, stable fibrotic scarring, solitary calcifications, old healed lesions, stable localized non-infectious anomalies, chronic interstitial fibrosis, chronic reticulonodular patterns without recent changes, stable chronic lesions without acute signs, or purely non-persistent anomalies.

Optimizer output at Step 50 (with Prior)
Reasoning:
The model missed some true cases of pneumonia and incorrectly identified some non-pneumonia cases. To improve the accuracy, refine the criteria to better identify pneumonia even in less obvious cases and differentiate between chronic stable conditions and acute infections.

New Model Descriptions: — Model parameters θ_{50}
The input is an X-ray image. To classify:
- Output 1 (Pneumonia) if there are lobar consolidations, asymmetric or bilateral ground-glass opacities, air bronchograms, moderate to significant alveolar fluid, asymmetric infiltrates, patchy inflammations, moderate to severe pleural effusion, central or peripheral acute opacities, air-fluid levels, increasing opacities, pneumatocele formations, rapidly changing patterns in chronic lesions indicative of acute infection, persistent cough-related findings, or recent onset of ground-glass opacities.
- Output 0 (No Pneumonia) if lung fields are clear, minimal localized fluid, no distinct acute opacities, stable fibrotic scarring, solitary calcifications, old healed lesions, stable localized non-infectious anomalies, chronic interstitial fibrosis, chronic reticulonodular patterns without recent changes, stable chronic lesions without acute signs, or absence of any acute infection markers.

Model parameters θ_{49} at Step 50 (No Prior)
1. If the image shows clear anatomical structures, uniform texture, symmetry, and normal lung fields without visible opacities, well-defined masses, streaks, irregular granular patterns, or any deviations, output class label 0.
2. If the image shows any distinct opacities or masses, interstitial patterns, visible fluid accumulations, asymmetry, irregular heart shapes, or multiple small opacities, output class label 1.
3. For granular or streaky opacities over any lung region, indistinct or uneven lung fields, subtle masses, minor deviations in symmetry, or any lung irregularities, output class label 1. For minimal irregularities or slight deviations, output class label 0.

Optimizer output at Step 50 (No Prior)
Reasoning:
The model misclassified several images where minor opacities or very light irregular patterns were present. We need to be more precise in detecting these subtle features while examining the overall lung fields more thoroughly to reduce false negatives for label 1. Specifically, minor irregularities or granular patterns that seem inconsequential should still be marked as potential abnormalities if they occur in specific regions, such as the periphery.

New Model Descriptions: — Model parameters θ_{50}
1. If the image shows clear anatomical structures, uniform texture, symmetry, and normal lung fields without any visible opacities, well-defined masses, streaks, or granular patterns, output class label 0.
2. If the image shows any distinct opacities or masses, interstitial patterns, visible fluid accumulations, asymmetry, irregular heart shapes, or multiple small opacities across the lung fields, output class label 1.
3. For granular or streaky opacities over any lung region, indistinct or uneven lung fields, subtle masses, minor deviations in symmetry, or any lung irregularities or granular patterns, including faint ones, in the periphery, output class label 1. For very minimal irregularities or slight deviations in central regions only, output class label 0.

(d) Model parameters after training

Figure 3 : PneumoniaMNIST image classification with and without prior.

Scan QR code to checkout our GitHub repo and paper!

Presentation at ICML 2024 Workshop on In-Context Learning

MAX PLANCK INSTITUTE FOR INTELLIGENT SYSTEMS — EBERHARD KARLS UNIVERSITÄT TÜBINGEN — UNIVERSITY OF CAMBRIDGE