LMPriors: pre-trained language models as task-specific priors

Kristy Choi*, Chris Cundy*, Sanjari Srivasta, Stefano Ermon
{kechoi,cundy}@cs.stanford.edu
*Equal contribution

Stanford | Stanford Artificial Intelligence

Outline

Overview

Method

Experimental Results

Limitations

Conclusion

Overview

Overview

- Choosing a suitable prior can greatly improve performance for low-data problems, but is difficult.
  - Generic priors (e.g. sparsity) do not take into account semantic information from variable names and metadata
  - Eliciting detailed priors from domain experts can be expensive, subjective, and time-consuming.
- Language Model Priors (LMPriors) is a method to find a task-specific, semantically aware prior from task metadata
  - We prompt a foundation model as a knowledge base.
  - LMPriors improves performance in settings as diverse as feature selection, causal inference, and safe reinforcement learning.

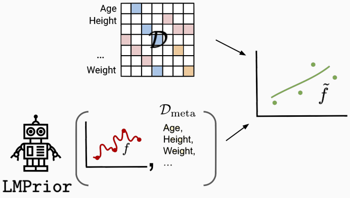


Figure 1: The LMPrior method. Metadata is combined with a large language model to provide priors which improve inferences from limited data.

Stanford Artificial Intelligence

Method

Problem Setting

- Learning task with dataset $\mathcal{D}$
  - Regression
  - Causal inference
  - Reinforcement Learning (offline)
- Learning procedure f
- Additional metadata $\mathcal{D}_{\text{meta}}$ is often present
  - Variable names
  - Variable descriptions
  - Data collection details

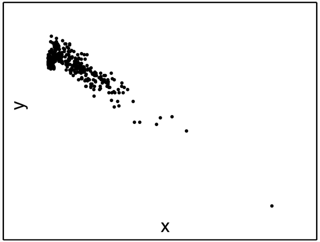


Figure 2: Data with no metadata

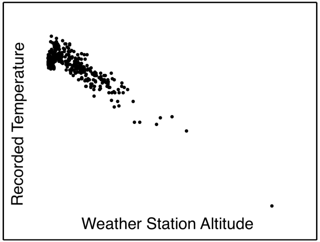


Figure 3: Data with metadata (variable names)

Method

- Represent the LMPrior framework with

$$\mathcal{P}_{\text{LMPrior}}(\mathcal{D}_{\text{meta}})(f) = \tilde{f}, \qquad \text{where}$$

- $\mathcal{P}_{\text{LMPrior}}$ is the LMPrior, consisting of a simple method to insert $\mathcal{D}_{\text{meta}}$ in the prompt of a foundation model and return a method to develop a new f.
    - The learning procedure f is the original learning procedure
    - The adjusted procedure $\tilde{f}$ is the new learning procedure incorporating the metadata.

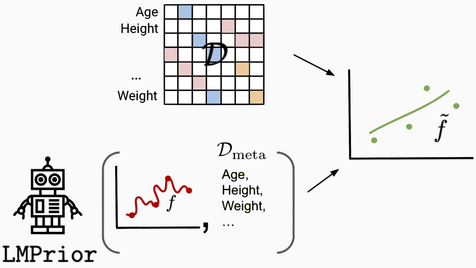


Figure 4: The LMPrior method. Metadata is combined with a large language model to provide priors which improve inferences from limited data.

Detailed Example: Causal Direction Inference

- Detecting causality direction, $\mathcal{D} = x_i, y_i$, $\mathcal{D}_{\text{meta}}$ is variable names and descriptions
- We formulate an LMPrior for this task by using the prompt template in figure ...
- We compute $\log P(\texttt{Variable A -> Variable B}) - \log P(\texttt{Variable B -> Variable A})$ and add this to the likelihood ratio given by a probabilistic test on the data $\mathcal{D}$.
- Add an explanation field to improve performance.

```
This is a set of causal relationship
facts.

A -> B means that A directly causes B.
The description explains why.
--
Variable A: Radiation
Description A: Radiation is the
average daily amount of ultraviolet
radiation
Variable B: Altitude
Description B: Altitude is the height
of a weather station
Context: The weather on Earth

Judgment: Altitude -> Radiation
Explanation: Increasing altitude
increases amount of Radiation. There
is no mechanism for Radiation to
change altitude

...
```

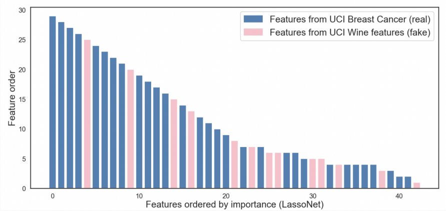
task description

example

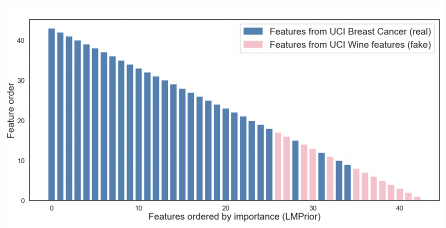
Figure 5: The prompt for the causal direction inference task.

Experimental Results

Feature Selection: Proof-of-Concept



(a) LassoNet Features



(b) LMPrior Features

Figure 6: Comparison of LassoNet with LMPrior on the feature separation task for the UCI Breast Cancer-Wine Quality dataset combination. Features are ordered according to importance. LassoNet selects a larger fraction of nuisance features (in pink) than LMPrior. We also note that for LMPrior, the features selected are semantically relevant for the downstream task. Some features returned by LassoNet are tied in importance.

Feature Selection: Census Data

	Random Forest	Logistic Regression	SVM	GBM
Subset	0.66	0.64	0.64	0.66
Full	0.74	0.94	0.95	0.86
Random	0.63 ± 0.05	0.62 ± 0.05	0.65 ± 0.16	0.60 ± 0.02
Lasso (C=0.001)	-	**0.95**	**0.96**	-
MRMR (*Radovic et al., 2017*)	0.73	0.75	0.73	0.71
LMPrior	***0.83***	**0.95**	**0.96**	**0.87**

Table 1: We train various methods to predict commute time from 286 variables included in the US Census Microdata, with different forms of inductive bias/regularization. Higher is better. LMPrior outperforms all baselines.

Safe Reinforcement Learning

- We use LMPriors in a Reinforcement Learning context as a form of reward shaping.
    - Describe the action about to be taken and elicit reward from the foundation model
    - This reward is added on to the reward obtained from the environment.
    - Foundation model gives negative reward for the agent getting closer to the hazardous water
- Have **2917** $\pm$ 85 safety violations during training for the LMPrior policy, against 8278 $\pm$ 1079 safety violations for the baseline.

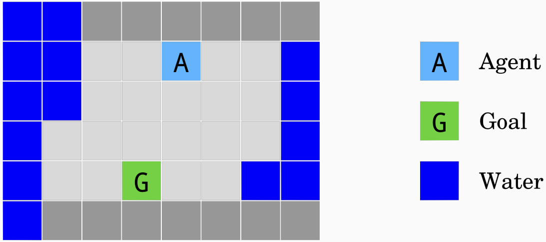


Figure 7: The Island Navigation gridworld as in (*Leike et al., 2017*). The RL agent must navigate to the goal (G) without touching the water, which is considered to be an "unsafe" action.

Stanford Stanford Artificial Intelligence

Limitations

Limitations

- Should be careful not to propogate harmful or toxic biases via LMPriors.
- Should be wary of prompt-hacking by trying several different prompts
  - Suggests standardized prompts
- Limited use where there is little semantic meaning to variables or large amounts of data

Conclusion

Conclusion

- Choosing a suitable prior can greatly improve performance for low-data problems, but is difficult.
  - Generic priors (e.g. sparsity) do not take into account semantic information from variable names and metadata
  - Eliciting detailed priors from domain experts can be expensive, subjective, and time-consuming.
- Language Model Priors (LMPriors) is a method to find a task-specific, semantic prior from task metadata
  - We prompt a foundation model as a knowledge base.
  - LMPriors improves performance in settings as diverse as feature selection, causal inference, and safe reinforcement learning

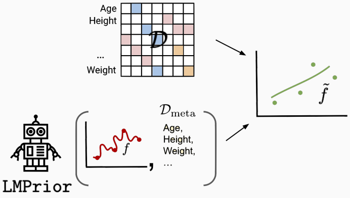


Figure 8: The LMPrior method. Metadata is combined with a large language model to provide priors which improve inferences from limited data.

Thanks