CS772: Project Proposal

Chaitanya Bramhapurikar
230305

Riyanshi
220903

Shubhanshu Mishra
221048

Naveen Kumar
220699

1 Objective

The objective of this project is to enhance AutoElicit by integrating LLM-generated priors into deep learning models, active learning strategies, and few-shot/zero-shot learning. We aim to improve sample efficiency, training stability, and data acquisition in low-data regimes. The project will compare AutoElicit with traditional methods to optimize learning and generalization.

2 Project Ideas

2.1 Enhancing AutoElicit with Active Learning for Optimal Data Collection

2.1.1 Idea:

Use LLM-generated priors to improve active learning strategies.

2.1.2 Research Questions:

- Can AutoElicit priors guide active learning strategies?
- How does active learning with priors compare to uncertainty or random sampling?
- What is the optimal balance between LLM knowledge and real data?

2.1.3 Methodology:

- Integrate AutoElicit priors into active learning frameworks such as uncertainty sampling and query-by-committee.
- Compare performance with traditional active learning methods by analyzing model accuracy, data efficiency, and computational costs.
- Conduct experiments on datasets with varying data availability, including simulated low-data scenarios and real-world sparse datasets.
- Optimize selection criteria by evaluating trade-offs between LLM-generated priors and newly acquired data, ensuring an adaptive learning strategy.

2.2 AutoElicit for Few-Shot and Zero-Shot Learning in Small Data Domains

2.2.1 Idea:

Apply AutoElicit to improve few-shot and zero-shot learning.

2.2.2 Research Questions:

- Can AutoElicit priors replace fine-tuning in few-shot learning?
- How does AutoElicit compare to meta-learning approaches like MAML ?

2.2.3 Methodology:

- Design experiments to compare AutoElicit priors with standard fine-tuning approaches using datasets such as Omniglot and Mini-ImageNet.
- Evaluate performance in few-shot and zero-shot settings by measuring classification accuracy, transfer learning effectiveness, and computational efficiency.
- Compare results with meta-learning methods like MAML by running controlled experiments with equal computational and data constraints.
- Analyze transferability of LLM-generated priors across different tasks, ensuring that they generalize well to unseen domains and problems.

2.3 Extending AutoElicit to Deep Learning Models

2.3.1 Idea:

Extend AutoElicit to deep learning models like MLPs, CNNs, and LSTMs.

2.3.2 Research Questions:

- Can LLMs generate informative priors for deep neural networks?
- How does AutoElicit compare to pretraining techniques in terms of sample efficiency?
- Does prior elicitation improve training stability in low-data regimes?

2.3.3 Methodology:

- Implement AutoElicit-generated priors in deep learning models by integrating the priors as weight initializations or Bayesian priors in probabilistic models.
- Compare training performance with and without priors using standard benchmarks such as CIFAR-10, MNIST, and IMDB sentiment analysis.
- Evaluate sample efficiency by analyzing convergence speed, data utilization, and performance metrics such as accuracy and F1-score.
- Assess training stability by measuring variance in loss curves, robustness to hyperparameter changes, and the ability to generalize across different datasets.

References

- AutoElicit: Using Large Language Models for Expert Prior Elicitation in Predictive Modelling
- Understanding Uncertainty Sampling
- Query by committee
- Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks
- Medium blogs
- ChatGPT, DeepSeek