Problem Statement, Hypothesis, and Research Questions

Problem Statement: This project is concerned around refining the supervised learning model with prompt tuning, aiming to improve accuracy of the LLM by using a gate that predicts the quality/usability of soft prompts so that no further training is needed.

Current Hypothesis: By creating a gate that builds off prompt tuning, we can improve the accuracy of models without further fine-tuning.
Current Research Questions:

> Is there a relationship between sample perplexity and sample success/failure?
> Can we find a statistical representation of this relationship to improve the accuracy of prompt tuning?

Extended Literature Review

Hu, J., Zhang, Z., Chen, G., Wen, X., Shuai, C., Luo, W., Xiao, B., Li, Y., & Tan, M. (2025, May 27). *Test-Time learning for large language models*. arXiv.org. https://arxiv.org/abs/2505.20633

In this paper, they propose Test Time Learning for LLMs, or TLM for short. They've observed that high-perplexity samples are more informative for model optimization than low-perplexity samples, so they created the Sample Efficient Learning Strategy which uses those high-perplexity samples for test-time updates. They found that TLM improves the performance of original LLM's by at least 20% on domain knowledge adaptation. The researchers used LoRA (Low-Rank Adaptation), as they found it was more effective at preventing catastrophic forgetting than full parameter updates.

It could be possible to use a perplexity-defined strategy like this to create a gate, where we filter out prompts of a certain threshold of perplexity into a different untrained model. This would mean that we are inferring that a higher perplexity has a correlation to a less accurate response. While they use LoRA, we could probably adapt their method for a prompt tuning approach.

Choi, J., Kim, J., Park, J., Mok, W., & Lee, S. (2023). SMoP: Towards Efficient and Effective Prompt Tuning with Sparse Mixture-of-Prompts. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 14306–14316. https://doi.org/10.18653/v1/2023.emnlp-main.884

In this article, the researchers propose a Sparse Mixture of Prompts (SMoP) as a new method of prompt-tuning which uses short soft prompts that maintains performance gains while improving efficiency. They achieve this by using a gating mechanism that trains multiple smaller soft prompts that handle different subsets of data rather than a single long soft prompt. They have shown through experiments that SMoP outperforms fine tuning while also being more efficient.

We could adopt a similar gate mechanism, where the gate trains itself how to filter between good and bad sample predictions, which would be a more elegant solution than the proposed solution detailed in my writing above because we wouldn't have to figure out the relationship between a good/bad sample and the training model, making this a more generalized solution.

High-Level Proposed Approach

I will be using several transformer-based models from HuggingFace, such as BERT and RoBERTa.
While my research isn't directly related to cybersecurity, the outcome should hopefully create an alternative LLM training method that is more accurate. To do this, I'm aiming to create some sort of logical gate, either based off of a statistical formula or a self-learning gate, that can filter out good and bad sample prompts.
Right now, I'm trying to find a statistical relationship between perplexity and success/failure, so I am recreating the results from the P-tuning v2 paper (https://arxiv.org/pdf/2110.07602) while logging the confidence levels and successes to try to find some relationship.

Experimental Design

Model	Reference	Highest Accuracy:	Method
Prompt Tuning Gate	This work	N/A	Prompt Tuning with Gate
StreamingLLM	Xiao et al., 2023	91.37% / Llama-2	Attention Sink with Sliding Window
Prompt Tuning	Liu et al., 2021	93.1% / CoNLL03 w/ FT, MPT-2	Prompt Tuning
Unsupervised Fine-Tuning	Ji et al., 2025	96.0% / Qwen2.5-Math-7B-Instruct w/ GSM8K	Unsupervised Prefix Fine-Tuning
ZeroTuning	Han et al., 2025	93.0% / Deepseek-R1-14B (Flash) w/ SST-2	Initial Token Prompt Tuning
Test Time Learning for LLMs (TLM)	Hu et al., 2025	90.96% / Llama3.2-3B-Instruct w/ GSM8K	LoRA with Perplexity Gate
Sparse Mixture of Prompts	Choi et al., 2023	94.6% / T5-base w/ CB	Prompt Tuning with SMoP

Reproduce One or More SOTA Baselines

I am in the process of recreating the results from the test scripts of the P-tuning v2 paper: https://arxiv.org/pdf/2110.07602
Currently, I have recreated two scripts:

1. BERT + COPA: 71.0, comparable to their 73.0
2. BERT + BoolQ: 73.73, comparable to their 75.8

While there is a discrepancy between the accuracy of their results and mine, I believe this is due to the difference of time when the results were reported. Since I ran these now, there may have been small package or database updates that have slightly changed the accuracy of the results.