1. Impact of prompt instruction:

- Zero- shot:

  "Task: Classify review as either 'high-star' or 'low-star'." This led to achieving higher precision (61.31) and recall (57.16), resulting in an F1 score of 56.11, as the model can easily understand the task at hand. In contrast, one of the prompts I tried used rules for classification without explicitly stating the task, which I hypothesize that it introduced ambiguity. Although it provided an example and classification rules, the absence of clear task direction reduces precision (55.26) and recall (52.55), resulting in a lower F1 score of 49.33. The comparison shows that direct task instructions improve model performance by providing clearer focus, whereas rule-based prompts without explicit task guidance leads to lower accuracy and effectiveness.

- One-shot:

  A clear prompt, combined with a non-biased example, helped guide the model's decision-making process and can significantly improve performance compared to zero-shot, where no examples are given. For instance, the one of the prompts provided direct "task" ("Task: Classify review as either 'high-star' or 'low-star'.") instruction with a clear example of a review and its classification, resulting in high precision (82.39) and recall (71.39), with a solid macro F1 score of 72.72. In contrast, the second prompt uses classification rules without explicitly stating the task, leading to a drop in both precision (66.30) and recall (55.73), and a lower F1 score of 52.00. However, both prompting instances still performed better than the zero-shot due to the one-shot prompt including an example.

- Few-shot:

  Despite the inclusion of examples, the results—precision (31.50), recall (50.00), and macro F1 (38.65)—are lower than in the zero-shot setting which was counterintuitive. I hypothesize that these results might be due:

1. Few-shot prompts causing overload, overwhelming the model if there are too many examples or if the examples are not clearly tied to the task.
2. In zero-shot, the model relies on its pre-trained knowledge, which can sometimes outperform few-shot settings if the prompt clarity is higher. Adding examples does not always enhance performance if the examples are not representative or the prompt structure is very ambiguous.

Next steps would involve doing additional more prompt engineering that are less ambiguous and clearly relating to the task at hand. In addition, the model was giving me different result for a few shot for different trials; however, before I submitted this assignment the above metrics are the performance I was getting at the moment. So, there is inconsistency of results from the model probably due to the lack of reproducibility scripts in the model repositories that can also contribute to the inconsistencies, as it makes it difficult to replicate the exact experimental setup.

2. Impact of class definition:

   Excluding ambiguous data—reviews that are difficult to classify as clearly high-star (1) or low-star (0)—impacts all three classifiers (zero-shot, one-shot, and few-shot). While this simplifies the dataset, it requires to reduce the test size to match the predictions size, which lead to lower evaluation metrics as the model has fewer opportunities to make correct predictions.

3. Impact of in-context sample selection:

   The impact of in-context sample selection varies across zero-shot, one-shot, and few-shot setups. In few-shot classification, selecting in-context examples manually or randomly from the training set had no noticeable effect on performance metrics. This consistency is likely due to ambiguity in the prompting structure or lack of result consistency by the model, as discussed earlier, which limits the model's ability to leverage the provided examples effectively. In contrast, for one-shot classification, the choice of example introduced a bias—

selecting a high-star or low-star review as the example influenced the model's predictions, skewing performance metrics towards one class. For zero-shot classification, in-context sample selection is irrelevant as no examples are provided, leaving the model's performance unchanged.

4. Impact of in-context sample order:

   For the few-shot classification, changing the order of in-context examples (sample order) had no noticeable effect on performance metrics. This might unfortunately be due to the ambiguity in the prompting structure or lack of results consistency by the modal, limiting the ability to study the reordering of the provided examples. I hypothesize that the lack of clarity in the prompt may have overshadowed any potential impact of reordering, resulting in similar performance regardless of the sequence. For zero-shot and one-shot setups, sample order is not applicable, as there are no multiple in-context examples to reorder.

5. Performance(as noted on the link given)

Evaluation Method	**Precision (%)**	**Recall (%)**	**Macro F1 (%)**
Zero-Shot	61.31	57.16	56.11
One-Shot	82.39	71.39	72.72
Few-Shot (3 examples)	31.50	50.00	38.65

***Performance vary as examples are chosen randomly and the used model from hugging face lacks consistency of results due to lack of reproducibility scripts in the model repositories.

Based on the analysis results, large language model (LLM) performance varies significantly depending on the evaluation method and the number of examples provided in the prompt. In the zero-shot setup, the model achieved moderate scores (Precision: 61.31%, Recall: 57.16%, Macro F1: 56.11%) as it relied solely on pre-trained knowledge to perform the task without specific examples. The one-shot evaluation demonstrated the highest performance (Precision: 82.39%,

Recall: 71.39%, Macro F1: 72.72%) by providing the model with a single, well-chosen example that effectively guided its predictions. Conversely, the few-shot setup with three examples resulted in the poorest performance (Precision: 31.50%, Recall: 50.00%, Macro F1: 38.65%), likely due to ambiguities in the prompt or the inability of the model to generalize effectively from multiple examples.

A key factor influencing these results is the randomness of example selection during evaluation. The inconsistency in model outputs is worsened by the lack of reproducibility scripts in the Hugging Face model repositories, making it difficult to ensure repeatability across different runs. This limitation highlights the need for better prompt clarity and structured evaluation processes to improve model reliability and interpretability.