Part 1:

Q1: What is the key benefit of fine-tuning a pre-trained model?

A: A. It reduces the need for computational resources

Q2: Which of the following tools optimizes model deployment in Azure?

A: A. ONNX Runtime

Potential Tasks for fine-tuning:

1. Summarizing legal documents:
   1. I would use pre-trained GPT models
   2. Fine-tuning benefits this task because the GPT model is already trained to work with text. The only thing fine-tuning needs to do is train the GPT model on the nuances, complexities, and structure of legal documents
2. Classifying News Articles
   1. I would use pre-trained BERT models for this task
   2. Classifying news articles benefits from fine-tuning because it allows pre-trained models to adapt to news articles without providing lots of training data. The model is also trained on the nuances of different categories of news articles, allowing the model to more accurately determine a news article’s category.
3. Generating a poem
   1. I would use a pre-trained GPT model for this task.
   2. Generating poems benefits from fine-tuning because the user can adapt the model to only produce specific kinds of poems, such as poems written by specific people or from specific countries.

Part 2:

Task: Summarizing novels

Model used: t5-base

How I would evaluate model’s performance after fine-tuning:

After fine-tuning the t5-base model with summarizing novels in mind, I would first use the ROGUE (Recall-Oriented Understudy for Gisting Evaluation) evaluation since it is best suited for measuring text summarization. ROUGE measures the quality of he generated summaries, so I think this metric would be good to start with. Since ROUGE covers precision, recall, and f1, I do not need to run those metrics separately. If the novel dataset included summaries of every novel in the dataset, I would then check for accuracy to see how well the generated summaries match the summaries provided in the dataset, if there are any. I think the biggest challenge I will face with evaluating generated novel summaries comes from the length of both the summaries and the novels themselves. While novels are unlikely to have more than 1,000 pages, From what I researched, novel lengths can vary from 50,000 to 100,000 words. Because of this, novels are likely to exceed t5-base’s maximum sequence length of 512. Exceeding the maximum sequence length will lead to bad outputs, so the input novel text would need to be modified in some way, such as splitting the novel into multiple chunks or keyword extraction to ensure the generated summary successfully captures the input novel’s basic outline.

Part 3:

Q1: Fine-tuning eliminates the need for evaluation metrics. False

Q2: Azure Machine Learning provides tools for real-time monitoring of deployed models. True

Evaluating a fine-tuned model with evaluation metrics is just as important as evaluating a model trained from scratch. This is because fine-tuned models can still run into the same problems that affect trained-from-scratch models. One potential problem that both types of trained models can encounter is overfitting, because fine-tuned models can also become too accustomed to the data used to train it and as a result perform poorly on new data. Evaluating using F1-score is still important for fine-tuned models because the dataset used to fine-tune the model can be imbalanced, ultimately leading to biased results. Cross-validation is also important for evaluating fine-tuned models because the fine-tuned model’s ability to generalize to new data is still good to know. If the fine-tuned model suffered from poor generalization, it could be a symptom of overfitting. Fine-tuned models can still suffer from underfitting if the training data used for fine-tuning is insufficient for the model to adjust itself to the specific task.