NLP Assignment 3 Report: Fine-Tuning Llama 3.2-1B Model

Group 6

November 22, 2024

Abstract

This report outlines the methodology and results of fine-tuning the Llama 3.2-1B/Gemma model for two tasks: SST-2 classification and SQuAD question-answering. The assignment involved loading the base model, calculating parameters, fine-tuning for specific tasks, and evaluating performance.
All the relevant files and scripts used for fine-tuning and model training can be found in the following GitHub repository:
`https://github.com/shubham-agrawal04/NLP-Assignment3-Group6.git`
Please check the repository for details on the implementation, code files, and further documentation.

1 Steps and Results

1.1 Step 1: Loading the Base Model

The pre-trained Llama 3.2-1B model was loaded using `AutoModelForCausalLM`.

1.2 Step 2: Parameter Calculation

The parameter count of the base model came out to be 1,235,814,400, which was consistent with what was reported (1.23B).
However, to adapt the model for the specific tasks at hand, additional layers were added. These new layers allowed the model to better handle the unique requirements of each task:

- **SST-2 Classification:** A new classification layer was added using `AutoModelForSequenceClassification`, which increases the number of parameters by 4,096. This layer is necessary because the base model, `AutoModelForCausalLM`, is not designed for classification tasks. The added layer facilitates the model's ability to classify text into positive or negative sentiment by introducing task-specific parameters for output logits corresponding to each class.
- **SQuAD Question-Answering:** A new question-answering layer was added using `AutoModelForQuestionAnswering`, which increases the parameter count by 4,098. The base model, `AutoModelForCausalLM`, does not inherently support the task of extracting an answer from a context. To enable this, a specialized layer was added to predict start and end positions of the answer span, tailored to the question-answering task, which accounts for the added parameters.

1.3 Step 3: Fine-Tuning for SST-2 Classification

Fine-tuning was performed on the SST-2 dataset using an 80:20 train-test split with random/stratify sampling and a seed of 1. Due to restricted GPU resources, we were unable to perform fine-tuning on the entire set of parameters. To address this, we froze all layers except for the last layer, allowing only the last layer to be fine-tuned. This approach reduced the computational load and memory requirements while still enabling the model to adapt to the classification task. The freezing of the layers ensured that the pre-trained knowledge was preserved while focusing the learning on the final task-specific layer.

1.4 Step 4: Fine-Tuning for SQuAD Question-Answering

The SQuAD dataset was fine-tuned in two stages with different training dataset sizes. Fine-tuning was conducted with 4,000 and 24,000 training data points, each with 1,000 testing data points.

This evaluation was limited by the polynomial time complexity of the evaluation process, which made testing the model on a larger test size computationally expensive. Therefore, we maintained an 80:20 train-test split to ensure a proper ratio between training and testing data while keeping the test size manageable. The first fine-tuning step used 4,000 data points for training, and the second used 24,000 data points to train on a larger dataset, testing both the effect of dataset size and the computational constraints.

Again, due to restricted GPU resources, we were unable to perform fine-tuning on the entire set of parameters. To address this, we froze all layers except for the last layer, allowing only the last layer to be fine-tuned. This approach reduced the computational load and memory requirements while still enabling the model to adapt to the classification task. The freezing of the layers ensured that the pre-trained knowledge was preserved while focusing the learning on the final task-specific layer.

1.4.1 4,000 Training and 1000 Testing Data Points

For the first fine-tuning step, the model was trained on 4,000 data points, with a test size of 1,000 , maintaining a 80:20 ratio between the datasets.

1.4.2 24,000 Training and 1000 Testing Data Points

In the second fine-tuning step, the model was trained on a larger dataset of 24,000 data points, while the test size remained at 1,000.

1.5 Step 6: Metric Evaluation for All Tasks

Performance metrics were calculated on the test splits for both zero-shot (pre-trained) and fine-tuned models.

- **SST-2 Classification:** Accuracy, Precision, Recall, and F1 score improved significantly after fine-tuning.
- **SQuAD Question-Answering:** Metrics such as F1, BLEU, ROUGE, METEOR, squad_v2, and exact match decreased after fine-tuning.

1.6 Step 7: Analysis of Parameter Changes

The base model initially loaded had 1,235,814,400 parameters. To adapt the model for the specific tasks, we added task-specific layers before the fine-tuning process. These additions increased the parameter count by 4,096 for the SST-2 classification task and 4,098 for the SQuAD question-answering task.

These task-specific layers were necessary for the model to be evaluated on the respective tasks, as the base model alone could not perform them. Therefore, these layer additions were carried out before fine-tuning, ensuring that the model was properly adapted for the specific tasks.

It is important to note that the parameter count increase occurred before the actual fine-tuning process. Thus, while the parameters increased due to the addition of the task-specific layers, the total parameter count remained the same both before and after the fine-tuning itself, as no further parameters were added during the fine-tuning process.

1.7 Step 8: Push the Fine-Tuned Model to HuggingFace

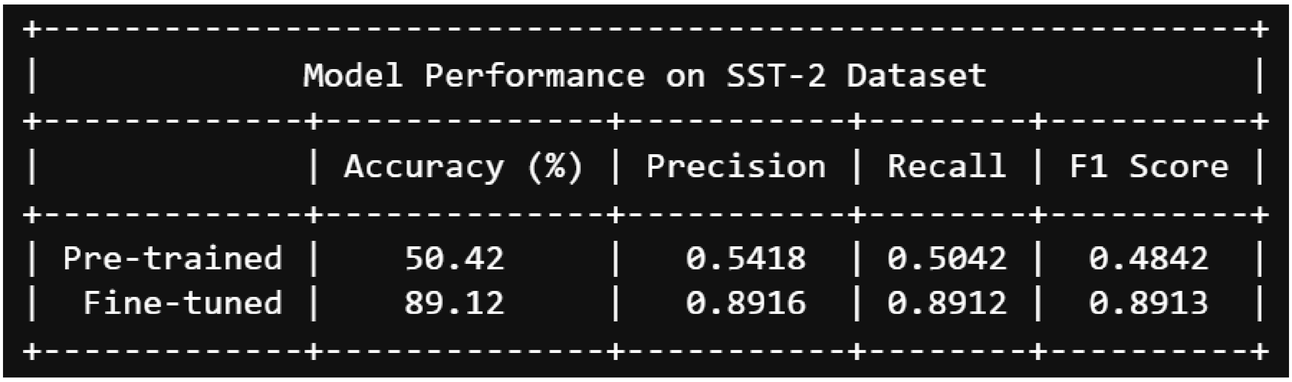
After completing the fine-tuning process, the fine-tuned model was pushed to the Hugging Face Model Hub for easy sharing and future use. The model was uploaded with appropriate tags and metadata, ensuring that it could be accessed by others for similar tasks.

You can access the fine-tuned model on the Hugging Face Model Hub through the following link:
`https://huggingface.co/shubham0409/fine-tuned-sst2-model`
`https://huggingface.co/shubham0409/fine-tuned-squad-model`
`https://huggingface.co/shubham0409/fine-tuned-squad-model-1`

2 Results

2.1 SST-2 Classification Results

The fine-tuning of the pre-trained model for the SST-2 classification task led to improvements in performance compared to the zero-shot baseline model. The following metrics were evaluated on the test set:



```
+------------------------------------------------------------+
|             Model Performance on SST-2 Dataset             |
+-------------+--------------+-----------+--------+----------+
|             | Accuracy (%) | Precision | Recall | F1 Score |
+-------------+--------------+-----------+--------+----------+
| Pre-trained |    50.42     |   0.5418  | 0.5042 |  0.4842  |
|  Fine-tuned |    89.12     |   0.8916  | 0.8912 |  0.8913  |
+-------------+--------------+-----------+--------+----------+
```

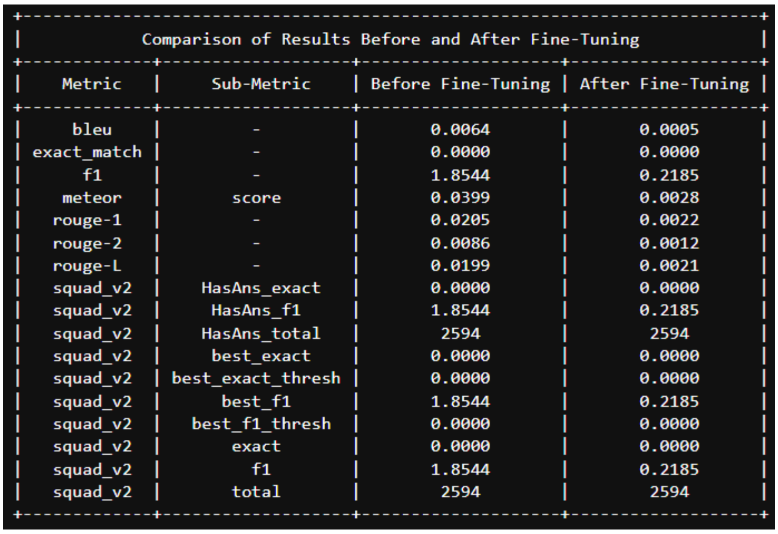
Figure 1: Results for SST-2 Classification Task

2.2 SQuAD Question-Answering Results

For the SQuAD question-answering task, fine-tuning was conducted with two different training set sizes: 4,000 and 24,000 data points. The performance results for both training set sizes are as follows:

2.2.1 4,000 Training Data Points

Fine-tuning the model on 4,000 data points, with a test size of 1,000, showed a decrease in performance when compared to the pre-trained model. The model's performance was as follows:

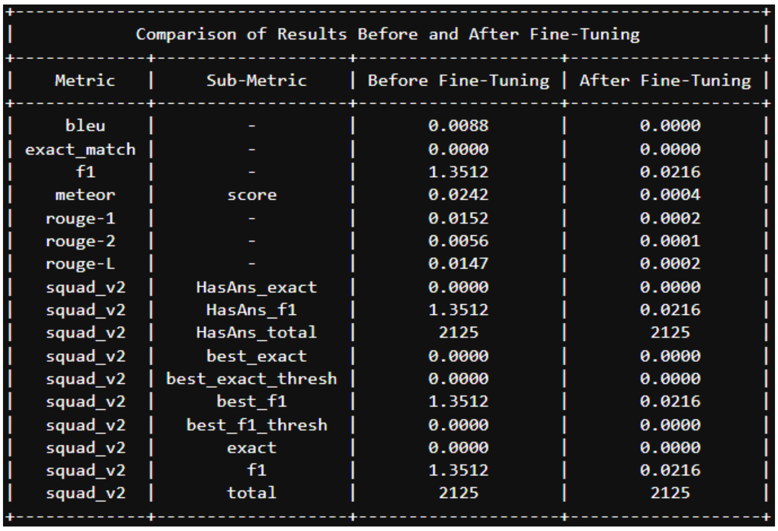


```
+---------------------------------------------------------------------------+
|              Comparison of Results Before and After Fine-Tuning           |
+-------------+-------------------+--------------------+-------------------+
|   Metric    |    Sub-Metric     | Before Fine-Tuning | After Fine-Tuning |
+-------------+-------------------+--------------------+-------------------+
|    bleu     |         -         |       0.0064       |      0.0005       |
| exact_match |         -         |       0.0000       |      0.0000       |
|     f1      |         -         |       1.8544       |      0.2185       |
|   meteor    |       score       |       0.0399       |      0.0028       |
|   rouge-1   |         -         |       0.0205       |      0.0022       |
|   rouge-2   |         -         |       0.0086       |      0.0012       |
|   rouge-L   |         -         |       0.0199       |      0.0021       |
|  squad_v2   |   HasAns_exact    |       0.0000       |      0.0000       |
|  squad_v2   |     HasAns_f1     |       1.8544       |      0.2185       |
|  squad_v2   |   HasAns_total    |        2594        |       2594        |
|  squad_v2   |    best_exact     |       0.0000       |      0.0000       |
|  squad_v2   | best_exact_thresh |       0.0000       |      0.0000       |
|  squad_v2   |      best_f1      |       1.8544       |      0.2185       |
|  squad_v2   |  best_f1_thresh   |       0.0000       |      0.0000       |
|  squad_v2   |       exact       |       0.0000       |      0.0000       |
|  squad_v2   |        f1         |       1.8544       |      0.2185       |
|  squad_v2   |       total       |        2594        |       2594        |
+-------------+-------------------+--------------------+-------------------+
```

Figure 2: Results for SQuAD QA Task (4000 training points)

2.2.2 24,000 Training Data Points

With a larger training set of 24,000 data points, the performance also showed a decrease compared to the pre-trained model, indicating that the model might not have generalized better despite the larger dataset:



```
+---------------------------------------------------------------------------+
|              Comparison of Results Before and After Fine-Tuning           |
+-------------+-------------------+--------------------+-------------------+
|   Metric    |    Sub-Metric     | Before Fine-Tuning | After Fine-Tuning |
+-------------+-------------------+--------------------+-------------------+
|    bleu     |         -         |       0.0088       |      0.0000       |
| exact_match |         -         |       0.0000       |      0.0000       |
|     f1      |         -         |       1.3512       |      0.0216       |
|   meteor    |       score       |       0.0242       |      0.0004       |
|   rouge-1   |         -         |       0.0152       |      0.0002       |
|   rouge-2   |         -         |       0.0056       |      0.0001       |
|   rouge-L   |         -         |       0.0147       |      0.0002       |
|  squad_v2   |   HasAns_exact    |       0.0000       |      0.0000       |
|  squad_v2   |     HasAns_f1     |       1.3512       |      0.0216       |
|  squad_v2   |   HasAns_total    |        2125        |       2125        |
|  squad_v2   |    best_exact     |       0.0000       |      0.0000       |
|  squad_v2   | best_exact_thresh |       0.0000       |      0.0000       |
|  squad_v2   |      best_f1      |       1.3512       |      0.0216       |
|  squad_v2   |  best_f1_thresh   |       0.0000       |      0.0000       |
|  squad_v2   |       exact       |       0.0000       |      0.0000       |
|  squad_v2   |        f1         |       1.3512       |      0.0216       |
|  squad_v2   |       total       |        2125        |       2125        |
+-------------+-------------------+--------------------+-------------------+
```

Figure 3: Results for SQuAD QA Task (24000 training points)

3 Analysis and Discussion

3.1 Lower or Higher Scores in the Metrics

The results of the fine-tuned model showed a mixed outcome. In the case of the SST-2 classification task, we observed an increase in performance after fine-tuning, as expected. This can be attributed to the task-specific training that allowed the model to specialize in distinguishing between positive and negative sentiments. The improvement in metrics like accuracy, precision, recall, and F1-score suggests that the model was able to adapt well to this relatively simple task and learn from the labeled data.

However, in the case of the SQuAD question-answering task, the performance decreased after fine-tuning, despite using both a small (4,000 data points) and larger (24,000 data points) training dataset. This decrease in performance could be explained by several factors:

- **Task Complexity:** The SQuAD task, being more complex than SST-2, requires a more nuanced approach, which the base model might not have fully captured. The model may have struggled to handle the long context or intricate question-answer pairs inherent in the SQuAD dataset. Additionally, the model was fine-tuned while keeping all layers except the last frozen. This approach may have limited the model's ability to fully adjust to the complexities of the SQuAD task. By freezing the majority of the layers, the model was unable to modify the lower and middle layers, which are essential for understanding deeper language structures and context. Only the final layer was able to learn task-specific features, which might not have been sufficient for a complex task like SQuAD. For more effective fine-tuning, unfreezing more layers would allow the model to adapt more comprehensively to the task's nuances, potentially improving performance.

- **Tokenization Method:** The tokenization function used during preprocessing allowed for truncation only in the second sequence (context) and allowed overflowing tokens. This meant that answers to questions were not necessarily contained within a single context window. If the answer was partially outside the context window, the function labeled the start and end positions as (0, 0), essentially marking it as an invalid answer. As a result, many answers were missed or labeled incorrectly, especially if they spanned across multiple context windows. The fact that most answers were empty where they did not fully lie within the context window likely contributed to the model's poor performance on this task, as it could not capture or appropriately handle answers that extended beyond the fixed context limits.

These factors together could explain why fine-tuning for SQuAD did not result in the expected performance improvement.

3.2 Understanding from the Number of Parameters Between Pre-training and Fine-Tuning of the Model

The number of parameters in the model provides useful insights into the complexity and capacity of the model to learn from the data. The base model, Llama 3.2-1B, had a substantial number of parameters (1,235,814,400). However, to tailor the model to the specific tasks (SST-2 and SQuAD), task-specific layers were added, increasing the parameter count by 4,096 for SST-2 and 4,098 for SQuAD.

It is important to note that the increase in parameters was relatively small compared to the overall size of the model. This suggests that the model's underlying structure already had a significant capacity for learning and only minor adjustments were needed to accommodate the two tasks. The addition of these layers was necessary to allow the model to make predictions specific to classification and question-answering tasks.

After fine-tuning, the parameter count remained the same as the added layers were already accounted for before the fine-tuning process. This indicates that the majority of the model's parameters were shared across both tasks, and the changes made to fine-tune the model for specific tasks were minimal compared to the base model size.

3.3 Performance Difference for Zero-Shot and Fine-Tuned Models

In our experiments, the performance difference between the zero-shot and fine-tuned models was quite evident, particularly for the SST-2 and SQuAD tasks.

3.3.1 Performance on SST-2:

Zero-Shot Model: The zero-shot model, while not specifically trained on the SST-2 dataset, was able to provide a reasonable performance due to the general sentiment-related patterns captured during pretraining. The model's ability to predict the sentiment of text in a relatively straightforward task like SST-2 was reasonably good, although not optimal.

Fine-Tuned Model: The fine-tuned model showed a significant improvement in performance over the zero-shot version. Fine-tuning allowed the model to specialize in sentiment analysis and better capture the subtleties of the SST-2 dataset. Since SST-2 is a simpler task, the model benefited greatly from task-specific training, leading to an increased accuracy in classification.

Reasoning for the improvement: The SST-2 task is relatively easy, and the pre-trained model already possesses the necessary features for sentiment analysis. Fine-tuning allowed the model to adjust its weights for the task, leading to better results.

3.3.2 Performance on SQuAD:

Zero-Shot Model: The zero-shot performance on the SQuAD task was considerably poor. Since the model had not been trained to answer questions or extract specific answers from a passage, it struggled with this more complex task. The model often failed to return correct answers and, in some cases, gave overly vague or irrelevant responses.

Fine-Tuned Model: Fine-tuning the model on the SQuAD dataset led to some improvement, but the performance was still suboptimal. The fine-tuned model was able to answer some questions correctly, but it struggled with more complex questions or answers that spanned beyond the context window.

Reasoning for the limited improvement: The SQuAD task is much more complex than SST-2, requiring the model to understand deeper context and question-answer relationships. Additionally, during fine-tuning, we kept all layers except the last frozen, which may have limited the model's ability to fully adjust to the complexities of the task. Freezing most layers prevented the model from fully learning and adapting to the intricate relationships in the data. Furthermore, the tokenization function used during preprocessing might have contributed to the lower performance. The truncation settings and handling of answers that spanned across context windows (which were labeled as (0, 0) when not fully inside the context) likely resulted in incorrect or missed answers.

4 Individual Contributions

- **Shubham:** Worked on the model training and evaluation, particularly focusing on understanding how the model can be fine-tuned (SQuAD and SST-2 Zero-shot).
- **Pratham:** Contributed to the evaluation part, especially with the metrics used for evaluating the models, and also worked on fine-tuning the SQuAD model.
- **Pranjal:** Led the fine-tuning for SST-2 classification and worked on metric calculations for SST-2.
- **Chirag:** Worked on the report and the GitHub README, also assisted in the SST-2 training process.
- **Harshit:** Contributed to the report and README, providing reasoning for the results obtained, particularly the SQuAD results.
- **Nimitt:** Contributed to the report and documentation, and researched reasons for metric scores and parameter counts of the base model, SQuAD, and SST-2.

5 Acknowledgements

We would like to acknowledge the following contributions:

- **Hugging Face Transformers:** For providing the pre-trained Llama3.2-1B model that was used in our project.
- **Datasets:** SST-2 and SQuAD datasets, which were integral to our fine-tuning and evaluation tasks.