# Running GenAI on Intel AI Laptops and Simple LLM Inference on CPU and fine-tuning of LLM Models using Intel® OpenVINO™

| Team Members | Email ids |
|---|---|
| Om Gadhvi | om.gadhvi16147@sakec.ac.in |
| Ritika Mayekar | ritika.mayekar16370@sakec.ac.in |

# Table of Contents

# Table of Figures

## Problem Statement

To run GenAI on Intel AI Laptops and Simple LLM Inference on CPU and fine-tuning of LLM Models using Intel® OpenVINO™

Purpose:
To demonstrate the capability of running GenAI on Intel AI Laptops by implementing a simple LLM inference on CPU and fine-tuning LLM models using Intel® OpenVINO™. The goal is to achieve an efficient and high-performing chatbot that leverages advanced AI technology while maintaining accessibility on consumer-grade hardware.

Objectives:

1. Create a responsive and intelligent chatbot.
2. Optimize the model using OpenVINO.
3. Develop an intuitive user interface.
4. Highlight the potential of compact AI models in practical applications.

# Methodology

The project was executed in five distinct phases to develop and deploy an Intel-optimized chatbot effectively.

1. **Model Selection:**

   Chose the Tiny Llama 1B model for its balance between performance and efficiency, considering model size, inference speed, and Intel hardware compatibility.

2. **Model Fine-tuning:**

   Used INT4 quantization techniques for compressing the Tiny Llama 1B model, employing asymmetric quantization where 80% of the tensors were quantized as groups of 128.

3. **Intel OpenVINO Optimization:**

   Leveraged the Intel® OpenVINO™ toolkit to further optimize the model by converting it to OpenVINO's Intermediate Representation (IR) format and applying specialized techniques to enhance inference speed on Intel CPUs. Performance improvements were benchmarked accordingly.

4. **Deployment:**

   The OpenVINO-optimized model was integrated into a Python notebook, establishing the necessary runtime environment on Intel AI Laptops. Extensive testing was conducted to ensure stability and performance.

This structured approach enabled the creation of a highly efficient chatbot optimized for Intel hardware, demonstrating the potential of advanced AI models on consumer-grade devices.

## Architectural Design

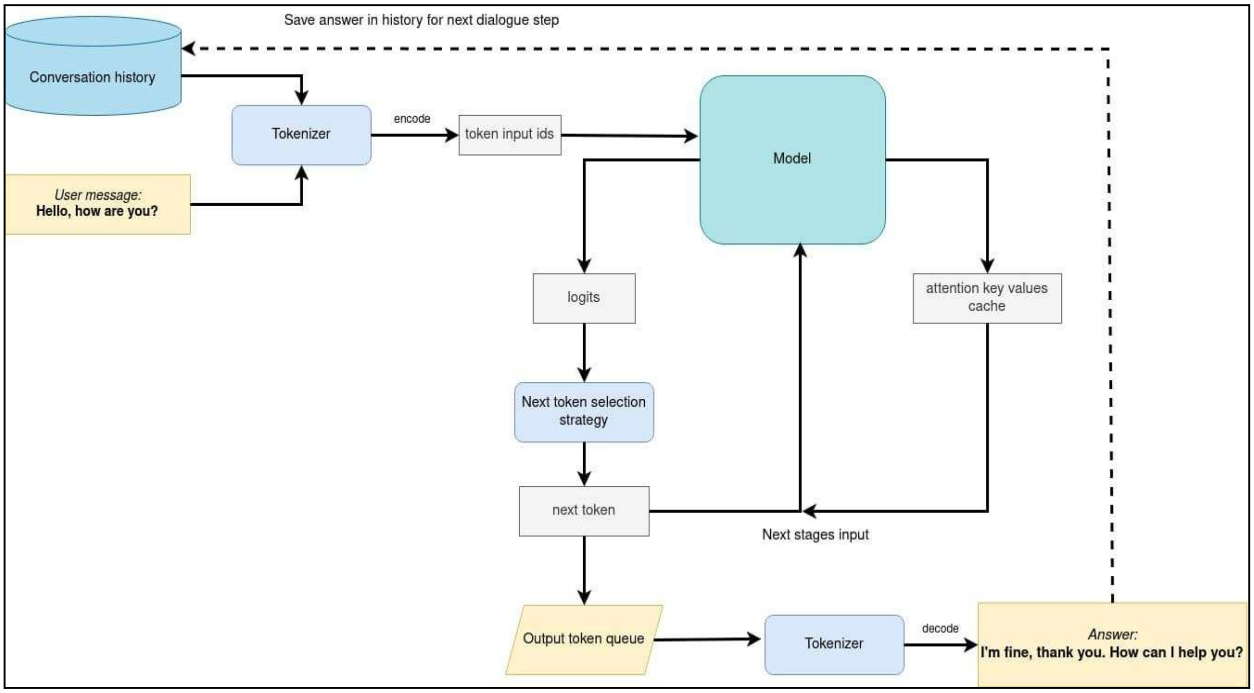
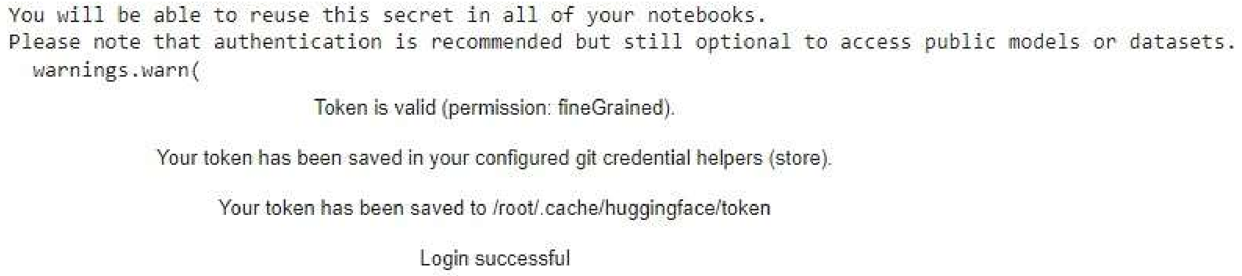


**Fig 1. Architecture Design**

The chatbot architecture centers on a streamlined pipeline optimized for instruction-following and contextual conversation. The system processes user input by combining the current question with previous conversation history, providing broader context for more accurate responses. The input is tokenized and fed into the Tiny Llama 1B model, fine-tuned and optimized using Intel® OpenVINO™ for efficient CPU-based inference on Intel AI Laptops.

The model generates token probabilities in logits format, from which the next token is selected based on the chosen decoding methodology. This process iterates until a complete response is generated, updating the conversation history to include both user input and model response. This cyclical approach maintains context across multiple interactions, enhancing response coherence and relevance over time.

# Results



```
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
```

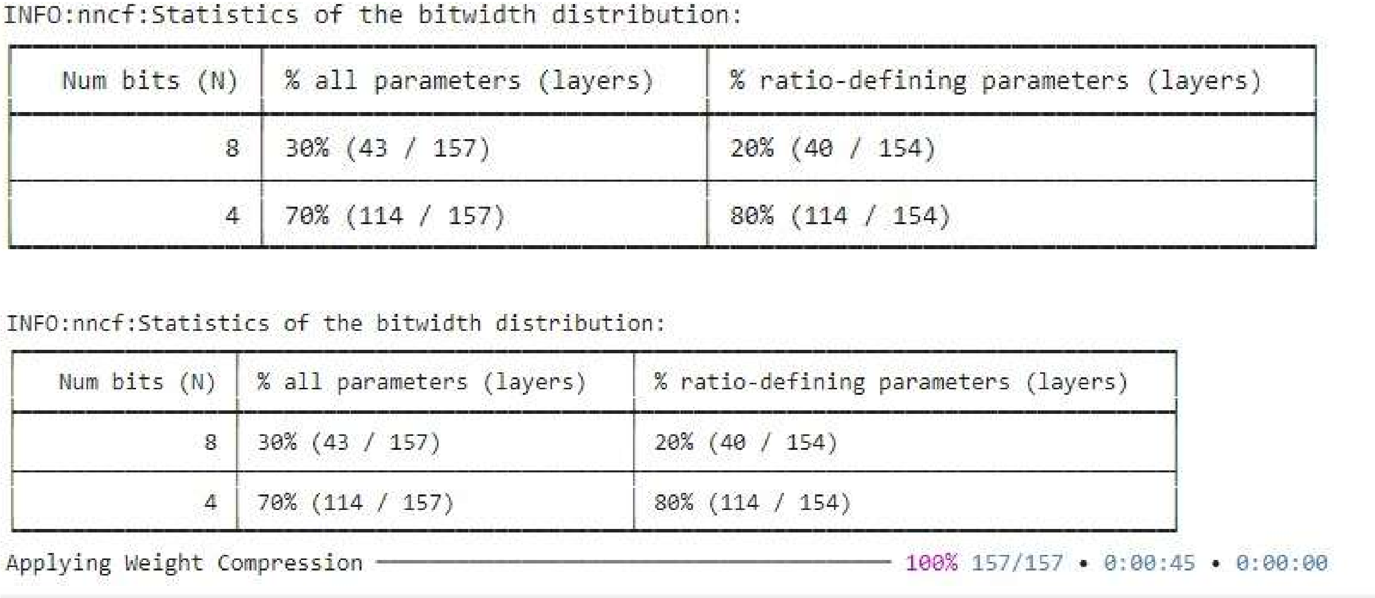
Token is valid (permission: fineGrained).

Your token has been saved in your configured git credential helpers (store).

Your token has been saved to /root/.cache/huggingface/token

Login successful

**Fig 2 : Authentication with the Hugging Face**

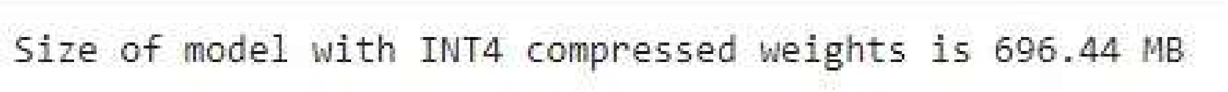


```
INFO:nncf:Statistics of the bitwidth distribution:
┍━━━━━━━━━━━━━━━━┯━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┯━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┑
│   Num bits (N) │ % all parameters (layers)   │ % ratio-defining parameters (layers)   │
┝━━━━━━━━━━━━━━━━┿━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┿━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┥
│              8 │ 30% (43 / 157)              │ 20% (40 / 154)                         │
├────────────────┼─────────────────────────────┼────────────────────────────────────────┤
│              4 │ 70% (114 / 157)             │ 80% (114 / 154)                        │
┕━━━━━━━━━━━━━━━━┷━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┷━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┙
```

```
INFO:nncf:Statistics of the bitwidth distribution:
┍━━━━━━━━━━━━━━━━┯━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┯━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┑
│   Num bits (N) │ % all parameters (layers)   │ % ratio-defining parameters (layers)   │
┝━━━━━━━━━━━━━━━━┿━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┿━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┥
│              8 │ 30% (43 / 157)              │ 20% (40 / 154)                         │
├────────────────┼─────────────────────────────┼────────────────────────────────────────┤
│              4 │ 70% (114 / 157)             │ 80% (114 / 154)                        │
┕━━━━━━━━━━━━━━━━┷━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┷━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┙
Applying Weight Compression ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100% 157/157 • 0:00:45 • 0:00:00
```

**Fig 3 : Statistics of the bitwidth distribution**

**Model Size**:



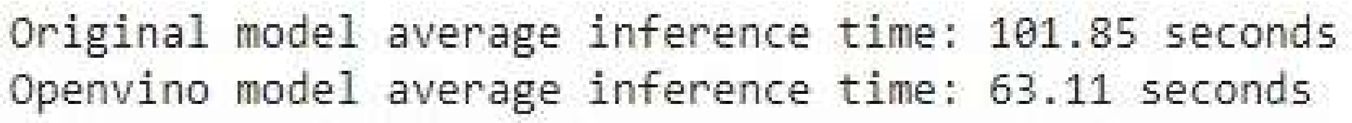
```
Size of model with INT4 compressed weights is 696.44 MB
```

**Fig 4. Output Size of Quantized Model**

- The original model had a size of approximately 2.2 GB.
- The OpenVINO™ optimized model was compressed to approximately 696.44 MB.

This represents a reduction in model size by approximately 69.02%. The reduction in model size through OpenVINO™ optimization demonstrates significant gains in memory efficiency without compromising the model's performance capabilities.

**Inference Time**



```
Original model average inference time: 101.85 seconds
Openvino model average inference time: 63.11 seconds
```

**Fig 5. Inference Time Results**

- The average inference time for the original model was **101.85 seconds**.
- The average inference time for the OpenVINO™ optimized TinyLlama-1.1B-Chat-v1.0 model was **63.11 seconds**.

This indicates that the OpenVINO™ optimized model was approximately 1.6 times faster than the original model in generating responses to the given prompts.

**Output Quality**:
Example of Model Outputs:

Prompt: "What's your favorite way to spend a weekend?"

- Original Model: "My favorite way to spend a weekend is with my family and friends. I love spending time together, trying new things, and enjoying each other's company. Some of my favorite weekend activities include going for a hike, watching a movie, playing board games, cooking a meal together, and having a picnic in the park. However, depending on the season and the location, there may be different activities that I enjoy more.."
- OpenVINO™ Model: "I do not have a personal preference for spending time on weekends. However, here are some common ways people enjoy a weekend:
  1. Going out with friends and having fun: this could involve going to a concert, watching a movie, trying new restaurants, or just hanging out at home.
  2. Reading a book or enjoying a good book club: reading can be a relaxing and enjoyable activity.
  3. Exploring new places: taking a road trip or exploring a new city or town can be an exciting and adventurous way to spend a weekend.
  4. Watching movies or playing games: watching movies or playing board games or card games can also be a fun way to spend a weekend.
  5. Spending time with family and loved ones: spending time with family and loved ones can be a special bonding experience.
  6. Cleaning up the house or doing household chores: cleaning up the house or doing household chores can be a great way to unwind and feel accomplished.
  7. Relaxing and unwinding: taking a bath, reading a book, or simply sitting by the pool"

The quality of the responses was assessed based on coherence, relevance, and completeness. The responses from both models were similar in terms of quality, though the OpenVINO™ model showed slight variations in phrasing.

# Conclusion

Our project successfully demonstrates the capability of running advanced GenAI applications on Intel AI Laptops, specifically through the implementation of an efficient and user-friendly chatbot. By leveraging the Tiny Llama 1B model, optimized with Intel® OpenVINO™ and fine-tuned using INT4 precision, we have shown that complex language models can perform effectively on consumer-grade CPUs.

The chatbot's architecture, combining an optimized inference pipeline showcases the potential for deploying sophisticated AI solutions on accessible hardware. This achievement not only highlights the power of Intel's AI optimization tools but also paves the way for more widespread adoption of AI in everyday computing scenarios.

Key outcomes of this project include:

1. Successful compression of the Tiny Llama 1B model to 696.44 MB while maintaining performance.
2. Efficient CPU-based inference using Intel® OpenVINO™ optimization.
3. Demonstration of the viability of running GenAI applications on consumer laptops.

These results underscore the potential for democratizing AI technology, bringing powerful language models to a broader audience without the need for specialized hardware. As we look to the future, this project serves as a stepping stone towards more accessible and integrated AI solutions in everyday computing environments.