**Simple LLM Inference on CPU and Finetuning of LLM Model to create a Custom Chatbot**

**PROBLEM STATEMENT:**

***This problem statement is designed to develop a simple and efficient system to perform inference using a Large Language Model (LLM) on CPU, ensuring it can deliver acceptable performance and have less response time.***

***Finetune a pre-trained Large Language Model (LLM)to create a custom chatbot for specific needs of users, ensuring it delivers accurate content, context-aware responses.***

**MAJOR CHALLENGES:**

***1. Pre-trained Language Models can have large file sizes, which may require significant memory to load and run.***

***2. Learn LLM inference on CPU.***

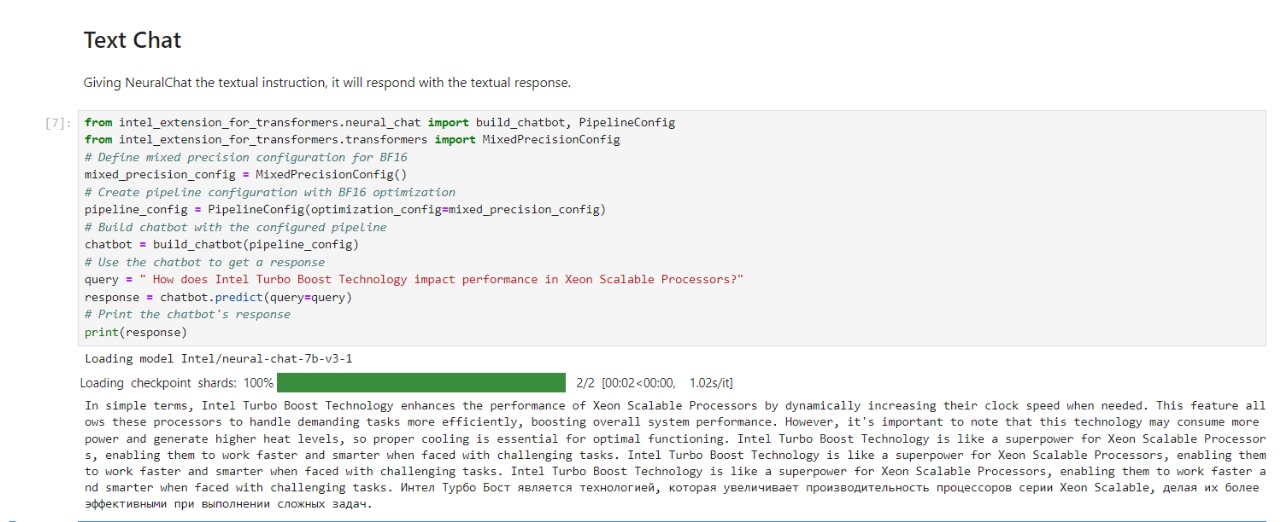
***3. Understanding the concept of fine-tuning and its importance in customizing LLMs.***

***4. Create a Custom Chatbot with fine-tuned pre-trained Large Language Models (LLMs) using Intel AI Tools.***

**SOLUTION:**

***These are the solution of our chatbot.***

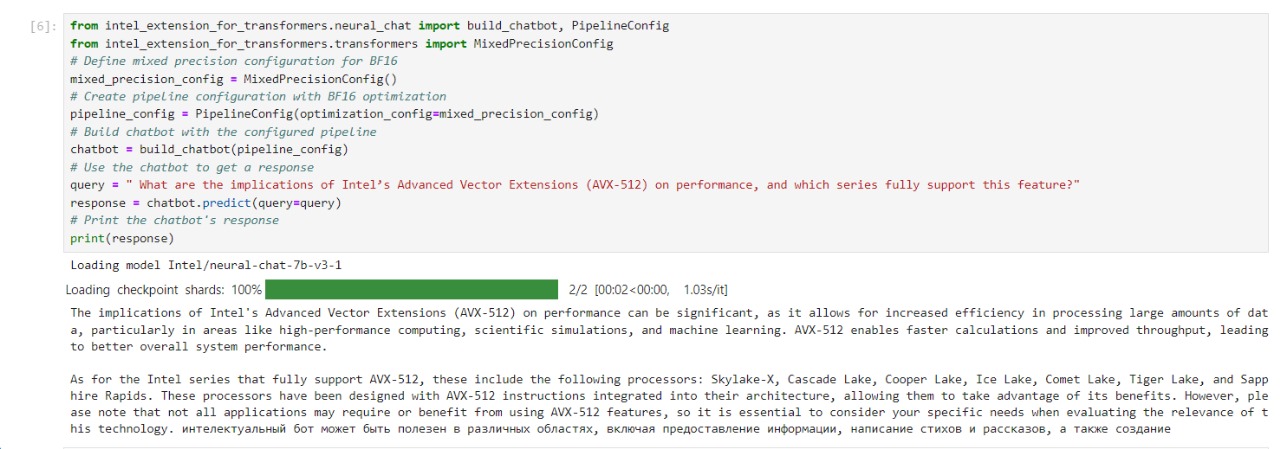
***1.How does intel turbo technology impact performance in Xeon scalable processors*?**

****

***TIME TAKEN: [00.02<00.00,1.02s/it]***

***Time taken for loading checkpoint shards is approx. 2 seconds (1.02 second per iteration for 2 iterations).***

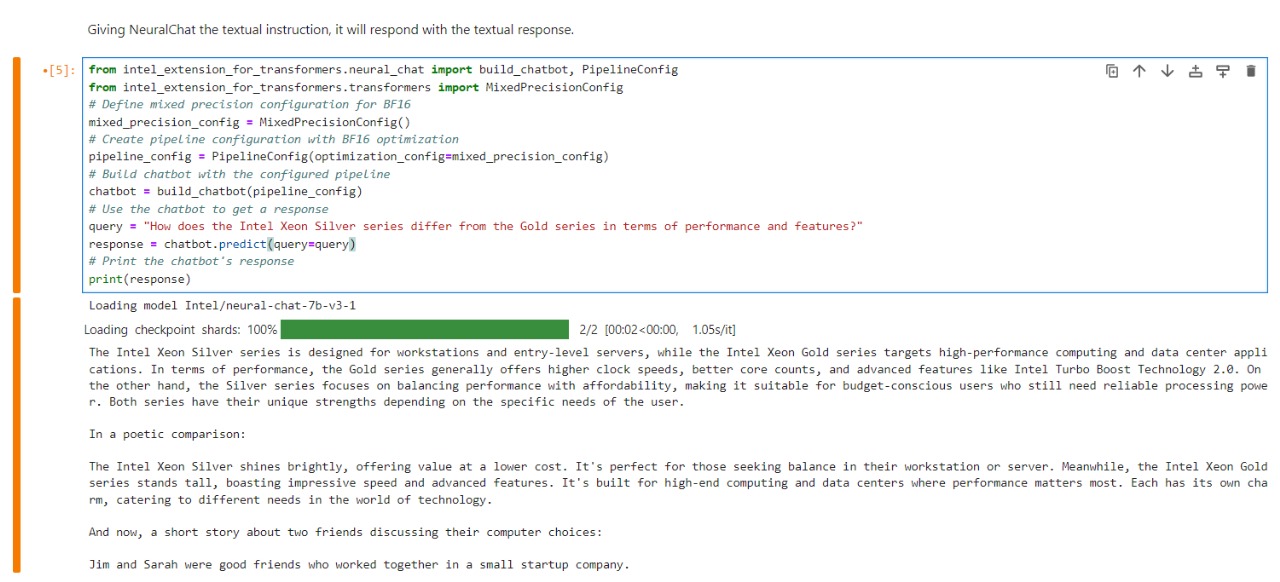
***2. What are the implications of intel’s advance vector extensions (AVX-512) on performance, and which series fully support this feature?***

****

***TIME TAKEN: [00.02<00.00, 1.03S/IT]***

***Time taken for loading checkpoint shards is approx. 2 seconds (1.03 second per iteration for 2 iterations).***

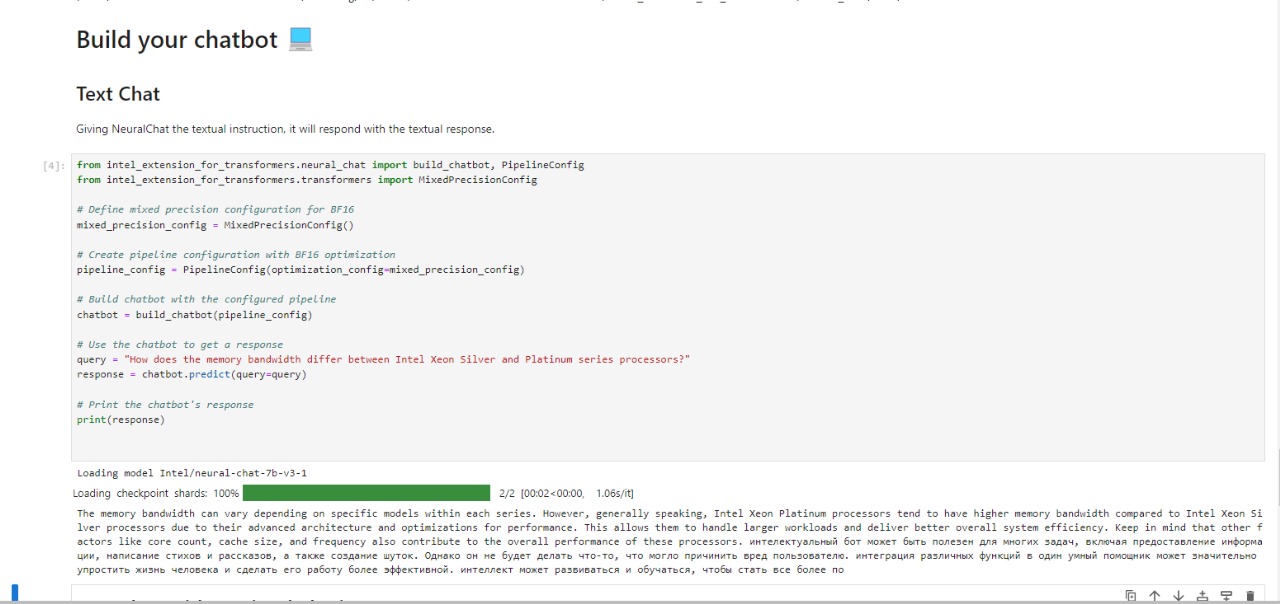
***3. How does the intel Xeon silver differ from the gold series in terms of performance and features?***

****

***TIME TAKEN: [00.02<00.00,1.05s/it]***

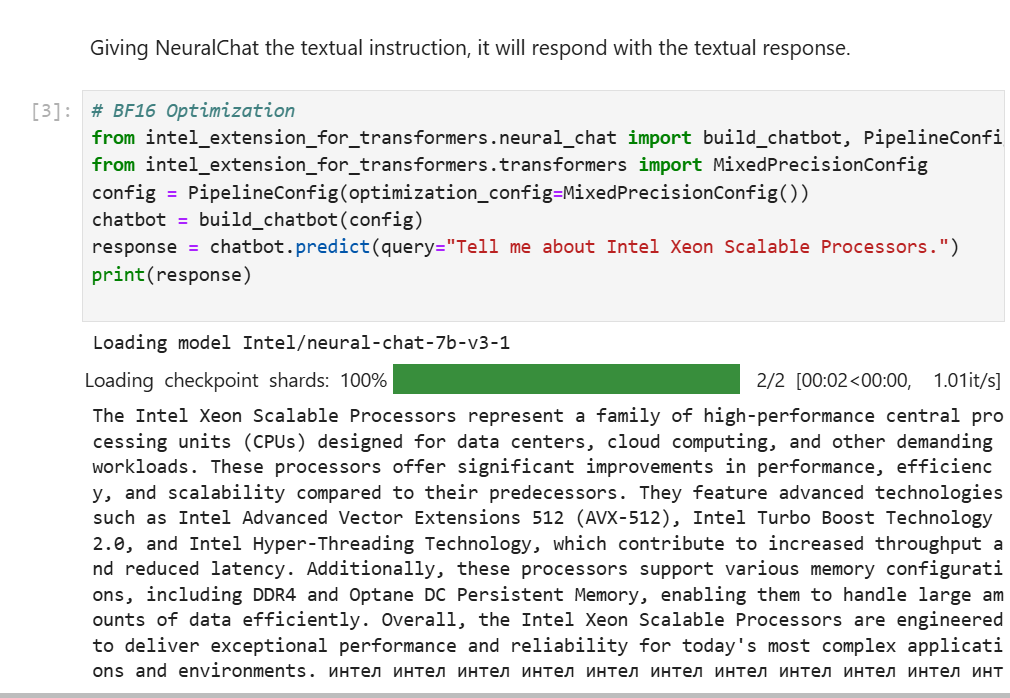
***Time taken for loading checkpoint shards is approx. 2 seconds (1.05 second per iteration for 2 iterations).***

***4. How does the memory bandwidth differ between intel Xeon silver and platinum series processors?***

****

***TIME TAKEN: [00.02<00.00, 1.06s/it]***

***Time taken for loading checkpoint shards is approx. 2 seconds (1.06 second per iteration for 2 iterations).***

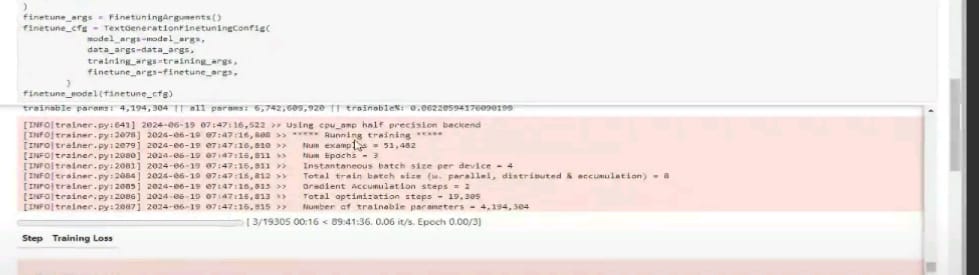
***5. Tell me about intel Xeon scalable processors*.**

***TIME TAKEN: [00.02<00.00, 1,01it/s]***

***Time taken for loading checkpoint shards is approx. 2 seconds (1.01 seconds per iteration for 2 iterations).***

**FINETUNING:**

***Finetuning help in modifying the model by making small changes to it.***

***For finetuning our model, we use intel extension for transformers.***

***Time Taken: [89:41:36]***

**FEATURES OFFERED:**

***IDC Intel Developer Cloud provide access to advanced intel technologies and tools in a flexible, cloud-based environment.***

***Intel AI analytics toolkit: Optimize and*** *accelerate* ***AI workloads with tools for data analytic and machine learning.***

* ***Ready to use environments for AI development.***
* ***Notebooks PyTorch and other.***
* ***Optimization and Interfaces.***

**PROCESS FLOW:**

***As our mentor suggest us to follow these particular steps to complete the chatbot and finetuning process. Steps are:***

**STEP1:**  git clone <https://github.com/intel/intel-extension-for-transformers.git>

**STEP2:** conda create -n itrex python-3.10 – y

**STEP3:**  conda activate itrex

**STEP4:** pip install intel-extension-for-transformers

**STEP5:** cd /intel-extension-for-transformers/intel\_extension\_for\_transformers/neural\_chat/

**STEP6:** pip install -r requirements\_cpu.txt

**STEP7:** pip install -r requirements.txt

**STEP8:** huggingface-cli login

**STEP9:** python3 -m pip install jupyter ipykernel

**STEP10:** python3 -m ipykernel install --name neural-chat-1 –user

*These are the steps that we followed to create environment for our chatbot and for finetuning it.*

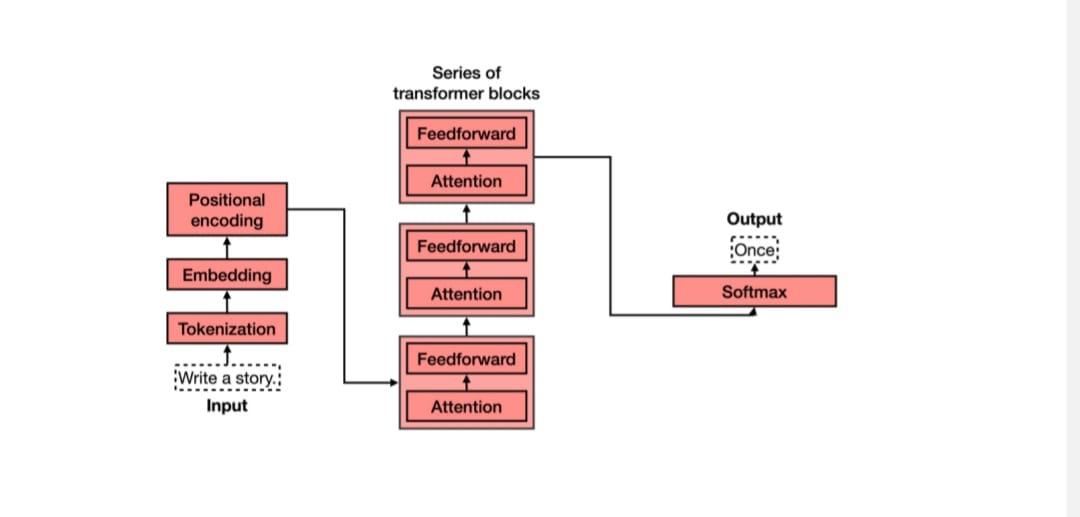
**ARCHITECTURE DIAGRAM:**

**TRANSFORMER MODEL:**

*A Transformer model consist of tokenization, embedding, positional encoding and series of transformer blocks and softmax***.**

**Transformer:**

*The transformer is a concatenation if many transformer blocks. Each one of these is composed by an attention component followed by a feedforward component (a neural network).*

****

**TECHNOLOGIES USED:**

*These are technologies that we used for our project.*

* *Jupyter Notebooks*
* *Intel DevCloud for the Edge*
* *Hugging Face*

**TEAM MEMBERS AND CONTRIBUTION:**

*Team Name: DAV University 2*

*Team Member: 5*

*Applicant Name: Muskan*

* *Team Member: Ankur*
* *Team Member: Garima Ojha*
* *Team Member: Twinkle*
* *Team Member: Sanjana Kumari*

*Our whole team have run the notebooks**individually and the screenshots are taken by Garima and Ankur (team members).*

*Finetuning is done by Twinkle and Sanjana (team members). Repository is done by Muskan(applicant). Report is made by Ankur after discussing the report format with the team.*

**CONCLUSION:**

*This training program help us to understand GEN AI and LLM (large language model). By working on project INTRODUCTION TO GEN AI AND SIMPLE LLM INFERENCE ON CPU AND FINETUNNING CHATBOT,*

*help us to explore more in this field and improve our knowledge.*

*A special thanks to INTEL UNNATI Team for providing this great opportunity and for supporting us in learning new skills and providing a platform where we can improve our skills***.**