

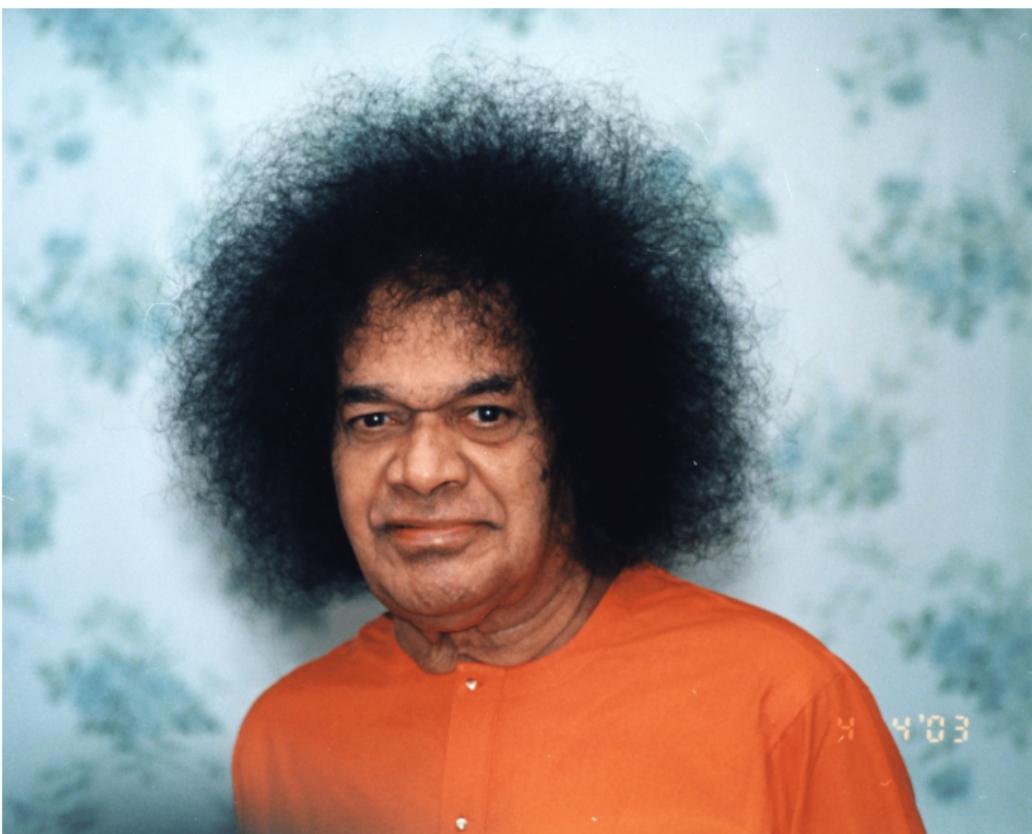
# BERT Base Uncased Inference Workload Characterization and Microarchitectural Exploration on Intel Skylake Processor

Vishwanath Saikiran Shetiya - 23017

Department of Mathematics and Computer Science  
Sri Sathya Sai Institute of Higher Learning

April 14, 2025





## **Offering at the Lotus Feet of Bhagawan Sri Sathya Sai Baba**

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

## 4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

## Background and Motivation

- Deep learning models are widely used in domains such as text, image, and speech processing.
  - These models are often computationally intensive, which limits their deployment on resource-constrained devices.
  - Therefore, it's essential to understand both the software and the underlying hardware.
  - This understanding helps identify bottlenecks, enabling optimization and efficient deployment for better performance.

## Problem Statement

- **Problem Statement 1:** Microarchitectural Characterization of BERT Base Uncased Model Inference and Exploring Microarchitectural Improvements.
  - **Problem Statement 2:** Identify bottlenecks at the Microarchitectural level and suggest improvement opportunities.

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

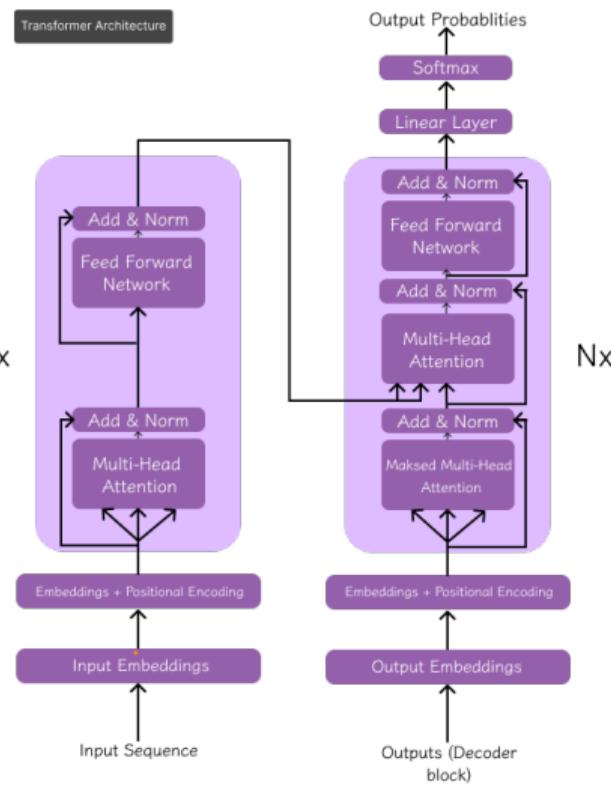
## 4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

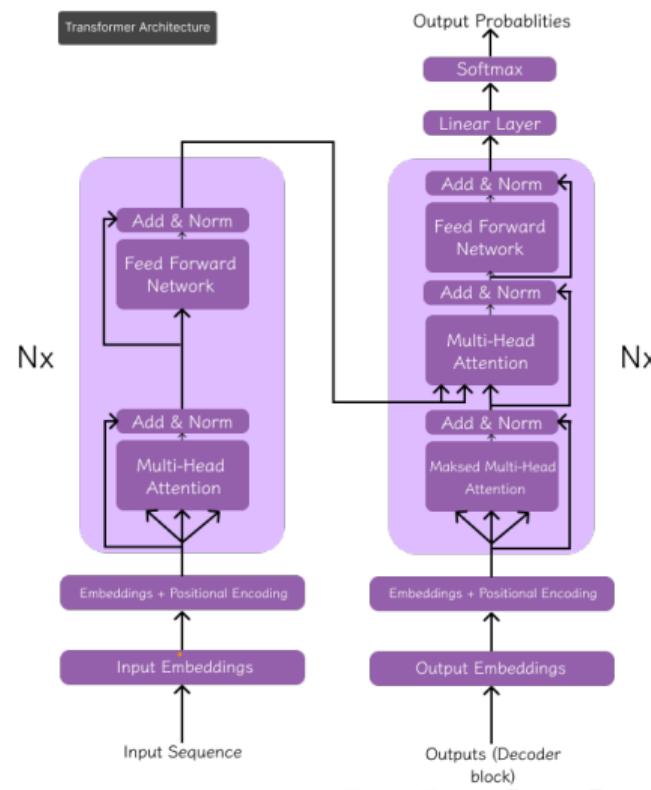
# Transformer - Attention is All You Need

- Introduced in 2017.
  - Neural network architecture based on a multi-head attention mechanism.
  - Used for various NLP tasks.
  - Used for image classification - Vision transformer.



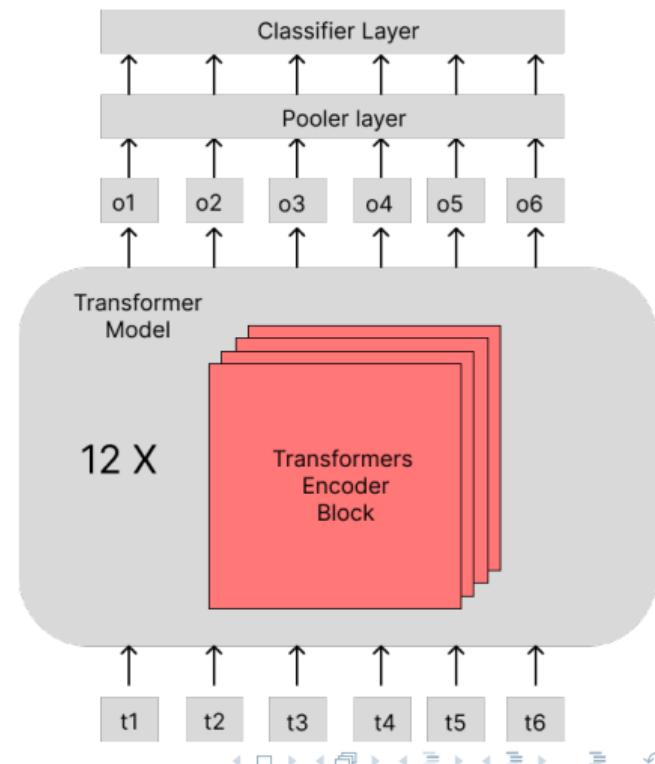
What distinguishes Transformer architecture from other LLM Architectures?

- **Self-Attention Mechanism**
    - Enables parallel processing, unlike RNNs and LSTMs, which are sequential.
    - Leads to faster training and inference.
  - **RNNs and LSTMs**
    - RNNs suffer from vanishing gradients, making training slow and unstable.
    - LSTMs introduced memory cells and gating mechanisms to reduce this issue.
    - However, they still process inputs sequentially, limiting speed.
  - **Transformers**
    - Use residual connections: input is added to the output of sublayers. Helps in better gradient flow and model convergence.



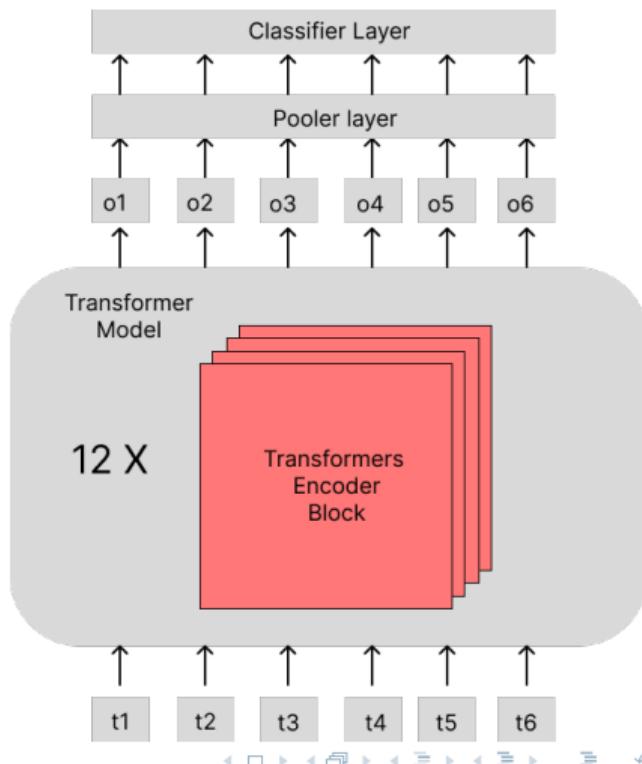
## BERT Base Uncased

- BERT - Bidirectional Encoder Representations from Transformers.
  - Introduced by Google in 2018.
  - Uses the transformer neural network architecture.
  - It is an encoder-based model.
  - Captures full context for better language understanding (Bidirectional).
  - **Different types:**
    - BERT Base (Cased and Uncased).
    - BERT Large (Cased and Uncased).
    - Others: Robustly Optimized BERT, SpanBERT, etc.



## Why BERT Base Uncased Model?

- Faster in training and inference.
  - Less computationally expensive.
  - For many NLP tasks, it is a starting point.
  - Since it is an uncased version, it helps with tasks that are case-insensitive.
  - Transfer learning.



## Micro-architecture of a Processor

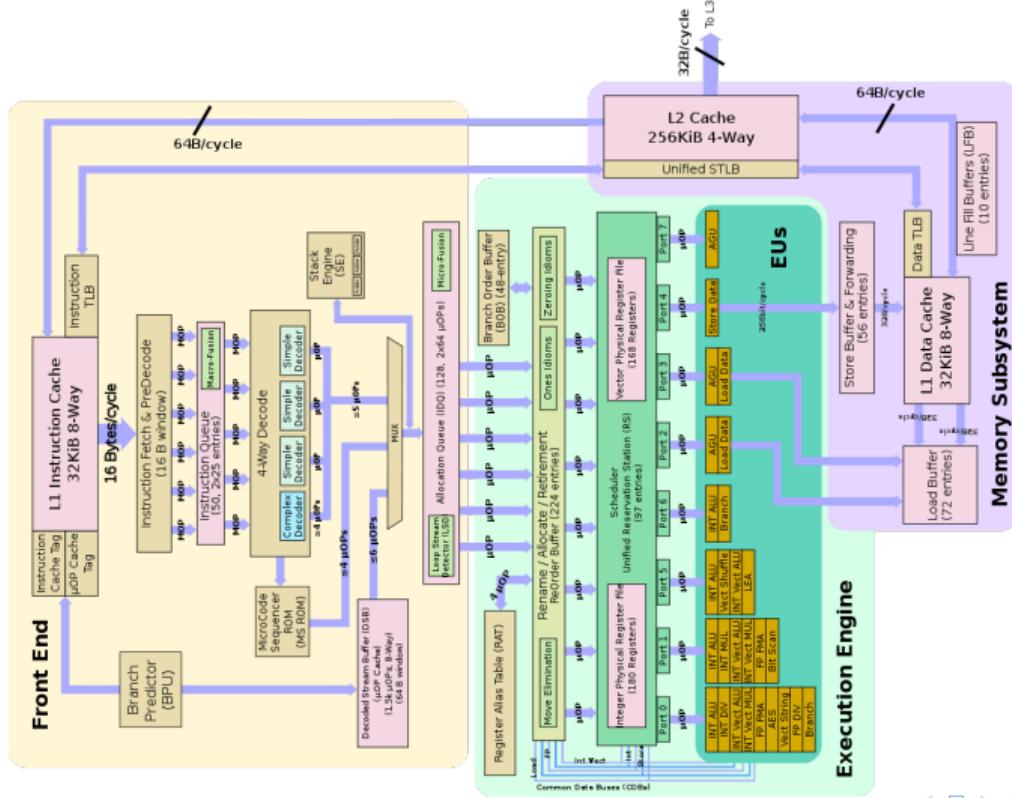
- **Computer Architecture** refers to the design of a processor including Instruction Set Architecture (ISA), supported data types, memory addressing, and I/O mechanisms.
  - **Microarchitecture**, also known as Computer Organization, deals with the internal implementation of the architecture.
  - It defines how the processor executes instructions including pipelining, execution units, cache hierarchy, branch prediction, and speculative execution.
  - While architecture is about **what** a computer can do, microarchitecture is about **how** it does it.

## Pipeline of Intel Core CPU

## Simple Five-Stage Pipeline

Instructions	Clock Cycles								
	1	2	3	4	5	6	7	8	9
Instruction X	IF	ID	EXE	MEM	WB				
Instruction X+1		IF	ID	EXE	MEM	WB			
Instruction X+2			IF	ID	EXE	MEM	WB		
Instruction X +3				IF	ID	EXE	MEM	WB	
Instruction X+4					IF	ID	EXE	MEM	WB

Intel Skylake Micro-Architecture



## Intel Skylake Micro-Architecture

- **4 per cycle:**
    - Front-End can allocate up to 4 uOps per cycle and Back-End can retire up to 4 uOps per cycle.
  - **Pipeline slots will be:**
    - Front-End
    - Back-End
    - Bad Speculation
    - Retirement

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

4 Preliminary Setup

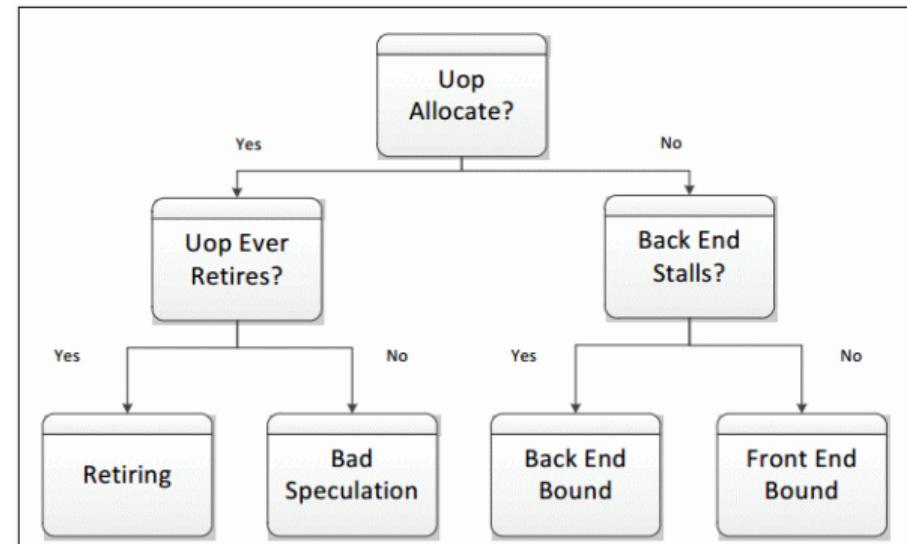
## 5 Observations and Results

## 6 Conclusions and Future Works

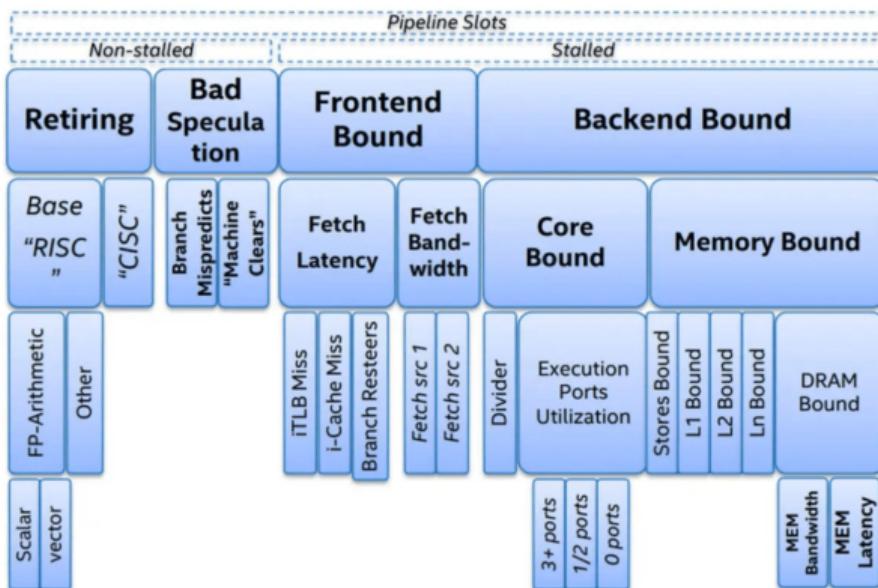
## Top-Down Methodology

- **Top-Down Analysis**

- It helps us to find the high-level causes of program execution.
  - Main categories are Front-End Bound, Back-End Bound, Retiring, or Bad Speculation.



## Top-Down Methodology



## Intel's VTune Profiler

- A performance profiling tool from Intel for analyzing code on Intel's CPU and GPU.
  - CPU Metrics: Cpu utilization, Cache misses, Port utilization.
  - Identify the Hotspots of our applications.

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

## System and Model Details

Parameter	Value
Processor	Intel(R) Core(TM) i5-8350U CPU @ 1.70GHz
Microarchitecture	Skylake
OS	Pop!_OS 22.04
Profiler	Intel VTune Profiler 2024.0.0
Framework	PyTorch
Model	BERT Base Uncased
Parameters	110M
Optimizer	AdamW
Number of Tokens (Input)	512
Hidden States	768

Table: System and Model Details

## Hyperthreading Disabled

- Threads compete for the same physical core resources.
  - Leads to contention and delays.
  - Hence, some operations will be bottlenecked.
  - `do_spin`: busy-waiting threads.
  - Threads constantly check for control.
  - Hotspot Analysis: consumes a lot of time.

Function	Module	CPU Time	% of CPU Time
[MKL_BLAS]@avx2_sgemm_k_emel_0	libmkl_avx2.so.2	35.953s	59.4%
do_spin	libgomp.so.1.0.0	4.173s	6.9%
[MKL_BLAS]@avx2_sgemm_s_copy_right4_ea	libmkl_avx2.so.2	1.942s	3.2%
[MKL_BLAS]@avx2_sgemm_k_emel_0_b0	libmkl_avx2.so.2	1.888s	3.1%
at::vec::AVX2::Vectorized<float>::exp_u20	libtorch_cpu.so	1.501s	2.5%
[Others]	N/A*	15.088s	24.9%

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

## 4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

## Microarchitecture Exploration

Top-Down Performance analysis of BERT Base Uncased Model - 1

Elapsed Time: 40.767s

Clockticks:	135,019,100,000
Instructions Retired:	382,173,600,000
CPI Rate ⓘ:	0.353
⦿ Retiring ⓘ:	69.0% of Pipeline Slots
⦿ Front-End Bound ⓘ:	6.1% of Pipeline Slots
⦿ Bad Speculation ⓘ:	1.5% of Pipeline Slots
⦿ Back-End Bound ⓘ:	23.5% of Pipeline Slots

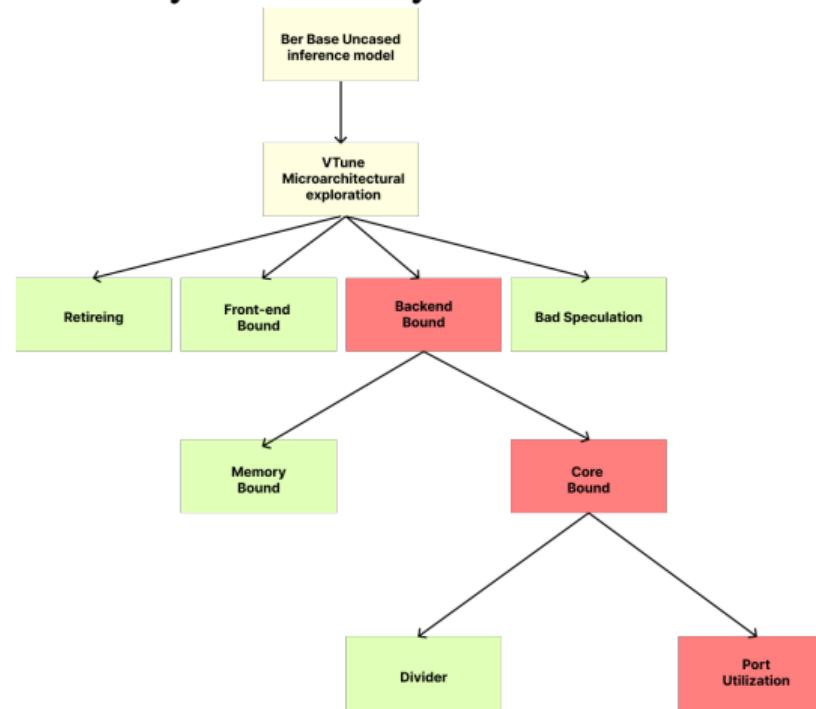
# Microarchitecture Exploration

## Top-Down Performance analysis of BERT Base Uncased Model - 2

Elapsed Time <small>⌚</small> : 40.767s <small>⬇️</small>	
Clockticks:	135,019,100,000
Instructions Retired:	382,173,600,000
CPI Rate <small>⌚</small> :	0.353
Retiring <small>⌚</small> :	69.0% of Pipeline Slots
Front-End Bound <small>⌚</small> :	6.1% of Pipeline Slots
Bad Speculation <small>⌚</small> :	1.5% of Pipeline Slots
Back-End Bound <small>⌚</small> :	23.5% <small>⬇️</small> of Pipeline Slots
Memory Bound <small>⌚</small> :	7.8% of Pipeline Slots
Core Bound <small>⌚</small> :	15.7% <small>⬇️</small> of Pipeline Slots
Divider <small>⌚</small> :	0.4% of Clockticks
Port Utilization <small>⌚</small> :	15.6% <small>⬇️</small> of Clockticks
Cycles of 0 Ports Utilized <small>⌚</small> :	4.3% of Clockticks
Cycles of 1 Port Utilized <small>⌚</small> :	5.4% of Clockticks
Cycles of 2 Ports Utilized <small>⌚</small> :	13.1% <small>⬇️</small> of Clockticks
Cycles of 3+ Ports Utilized <small>⌚</small> :	68.3% of Clockticks
Vector Capacity Usage (FPU) <small>⌚</small> :	100.0%
Average CPU Frequency <small>⌚</small> :	3.4 GHz
Total Thread Count:	8
Paused Time <small>⌚</small> :	0s

# Microarchitecture Exploration

## Top-Down Performance analysis Summary



## Hotspot Analysis

- Top Task - eltwise.
  - Helps in doing element-wise operations.
  - for our model in Gelu activation function,  
Layer normalization.

## Top Tasks

This section lists the most active tasks in your application.

Task Type	Task Time	Task Count	Average Task Time
eltwise	1.654s	48	0.034s

## Hotspot Analysis

- Hotspot function - General matrix multiplication, Takes about **60%** of the CPU time while running the BERT Base Uncased.
  - Which does:

$$C := \alpha \cdot \text{op}(A) \cdot \text{op}(B) + \beta \cdot C \quad (1)$$

Function	Module	CPU Time	% of CPU Time
[MKL BLAS]@avx2_sgemm_ker_nel_0	libmkl_avx2.so.2	26.002s	65.8%
[MKL BLAS]@avx2_sgemm_ker_nel_0_b0	libmkl_avx2.so.2	1.438s	3.6%
at::vec::AVX2::Vectorized<float>::exp_u20	libtorch_cpu.so	1.109s	2.8%
[MKL BLAS]@avx2_sgemm_scopy_right4_ea	libmkl_avx2.so.2	0.906s	2.3%
[MKL BLAS]@avx2_sgemm_ker_nel_nocopy_NN_b1	libmkl_avx2.so.2	0.838s	2.1%
[Others]	N/A*	9.252s	23.4%

# Port Utilization and Its Relation with the Hot Loop

## Ports on Skylake Processor

Skylake Execution Units and Ports							
Port 0	Port 1	Port 2	Port 3	Port 4	Port 5	Port 6	Port 7
INT ALU	INT ALU	AGU	AGU	STORE Data	INT ALU	INT ALU	AGU
INT DIV	INT MUL	LOAD Data	LOAD Data		VECT Shuffle	Branch	
INT Vect ALU	INT Vect ALU				INT Vect ALU		
INT Vect MUL	INT Vect MUL				LEA		
FP FMA	FP FMA						
AES	Bit Scan						
VECT String							
FP DIV							
Branch							

# Port Utilization and Its Relation with the Hot Loop

## Cycles of 2-Ports

- Represents the percentage of the total number of cycles where only 2 ports were utilized.
- running only 2 uOps per cycle on all execution port.
- But we issue 4 uOps per cycle.
- **Possible reasons for 2-ports utilization:**
  - Raw Data Dependencies.
  - Insufficient number of execution ports.
  - Vector Pipe Usage (100%) means that the floating point instructions are vectorized properly.

# Hotloop Pattern of Hotspot Function

Source	Assembly	Assembly grouping	Address	CPU Time	Instructions
Address ▲	Source Line		Assembly		
<b>Block 11:</b>					
0x59f040	vfmadd23ips	Ymm0, Ymm3, Ymm4	299.473ms	4,397.1	
0x59f045	vfmadd23ips	Ymm1, Ymm3, Ymm4	42.141ms	418.1	
0x59f04a	vfmadd23ips	Ymm2, Ymm3, Ymm12	321.889ms	3,998.1	
0x59f04f	prefetcht02	0x1800(%rbp)	26.899ms	260.1	
0x59f056	vbroadcasttsl	-0x7C(%rbp), Ymm3	335.223ms	4,360.1	
0x59f05c	prefetcht02	0x240(%rbx)	41.245ms	377.1	
0x59f063	vfmadd23ips	Ymm0, Ymm3, Ymm4	321.889ms	3,917.1	
0x59f068	vfmadd23ips	Ymm1, Ymm3, Ymm9	39.452ms	544.0	
0x59f06d	vfmadd23ips	Ymm2, Ymm3, Ymm13	345.201ms	3,313.1	
0x59f072	vbroadcasttsl	-0x7C(%rbp), Ymm3	61.867ms	681.1	
0x59f078	vfmadd23ips	Ymm0, Ymm3, Ymm4	300.370ms	2,641.1	
0x59f07d	vfmadd23ips	Ymm1, Ymm3, Ymm10	50.211ms	629.1	
0x59f082	vfmadd23ips	Ymm2, Ymm3, Ymm14	340.718ms	3,675.1	
0x59f087	vbroadcasttsl	-0x7C(%rbp), Ymm3	52.901ms	561.0	
0x59f09d	prefetcht02	0x280(%rbx)	294.093ms	2,298.1	
0x59f094	vfmadd23ips	Ymm0, Ymm3, Ymm7	21.513ms	405.1	
0x59f099	vmovupsy	-0x20(%rbx), Ymm0	401.688ms	2,255.1	
0x59f09e	vfmadd23ips	Ymm1, Ymm3, Ymm11	35.865ms	443.1	
0x59f0a3	vmovupsy	(%rbx), Ymm1	342.511ms	3,819.1	
0x59f0a7	vfmadd23ips	Ymm2, Ymm3, Ymm15	30.486ms	258.1	
0x59f0ac	vmovupsy	0x20(%rbx), Ymm2	364.927ms	4,530.1	
0x59f0b1	vbroadcasttsl	-0x7C(%rbp), Ymm3	21.513ms	377.1	
0x59f0b7	vfmadd23ips	Ymm0, Ymm3, Ymm4	305.792ms	3,920.1	
0x59f0bc	vfmadd23ips	Ymm1, Ymm3, Ymm6	30.486ms	350.1	
0x59f0c1	vfmadd23ips	Ymm2, Ymm3, Ymm12	373.893ms	4,522.1	
0x59f0e6	vbroadcasttsl	-0x8C(%rbp), Ymm3	37.658ms	566.1	
0x59f0cc	vfmadd23ips	Ymm0, Ymm3, Ymm5	316.509ms	3,998.1	

Source	Assembly	Assembly grouping	Address	CPU Time	Instructions
Address ▲	Source Line		Assembly		
<b>Block 12:</b>					
0x59f0d1	vfmadd23ips	Ymm1, Ymm3, Ymm9	111.182ms	1,026.1	
0x59f0d6	vfmadd23ips	Ymm2, Ymm3, Ymm13	318.302ms	4,158.1	
0x59f0db	vbroadcasttsl	-0x80(%rbp), Ymm3	51.108ms	498.1	
0x59f0e1	prefetcht02	0x2c0(%rbx)	295.887ms	3,590.1	
0x59f0e6	vfmadd23ips	Ymm1, Ymm3, Ymm10	59.177ms	831.1	
0x59f0e2	vfmadd23ips	Ymm2, Ymm3, Ymm14	402.585ms	4,702.1	
0x59f0f7	vbroadcasttsl	-0x64(%rbp), Ymm3	41.245ms	470.1	
0x59f0fd	vfmadd23ips	Ymm0, Ymm3, Ymm7	516.457ms	6,393.1	
0x59f102	vmovupsy	0x40(%rbx), Ymm0	23.312ms	229.1	
0x59f107	vfmadd23ips	Ymm1, Ymm3, Ymm11	636.398ms	7,166.1	
0x59f10c	vmovupsy	0x40(%rbx), Ymm1	23.312ms	205.1	
0x59f111	vfmadd23ips	Ymm2, Ymm3, Ymm15	702.048ms	8,373.1	
0x59f116	vmovupsy	0x80(%rbx), Ymm2	10.760ms	178.1	
0x59f11e	vbroadcasttsl	-0x80(%rbp), Ymm3	329.052ms	3,168.1	
0x59f124	vfmadd23ips	Ymm0, Ymm3, Ymm4	26.002ms	287.1	
0x59f129	vfmadd23ips	Ymm1, Ymm3, Ymm8	357.754ms	4,634.1	
0x59f12e	prefetcht02	0x300(%rbx)	13.449ms	209.1	
0x59f135	vfmadd23ips	Ymm2, Ymm3, Ymm12	379.273ms	4,363.1	
0x59f13a	vbroadcasttsl	-0x5c(%rbp), Ymm3	22.416ms	246.1	
0x59f140	vfmadd23ips	Ymm0, Ymm3, Ymm5	425.898ms	5,351.1	
0x59f145	vfmadd23ips	Ymm1, Ymm3, Ymm9	14.346ms	200.1	
0x59f14a	vfmadd23ips	Ymm2, Ymm3, Ymm13	292.300ms	3,489.1	
0x59f14f	vbroadcasttsl	-0x50(%rbp), Ymm3	16.139ms	197.1	
0x59f155	vfmadd23ips	Ymm0, Ymm3, Ymm6	427.691ms	5,241.1	
0x59f15a	vfmadd23ips	Ymm1, Ymm3, Ymm10	6.276ms	54.1	
0x59f15f	vfmadd23ips	Ymm2, Ymm3, Ymm14	403.482ms	4,278.1	
0x59f164	vbroadcasttsl	-0x54(%rbp), Ymm3	6.276ms	81.1	

Source	Assembly	Assembly grouping	Address	CPU Time	Instructions
Address ▲	Source Line		Assembly		
<b>Block 13:</b>					
0x59f190	vmovupsy	0x8(%rbx), Ymm2	132.701ms	1,778.1	
0x59f195	vbroadcasttsl	-0x80(%rbp), Ymm3	239.399ms	3,563.1	
0x59f19e	vfmadd23ips	Ymm0, Ymm3, Ymm4	121.941ms	1,649.1	
0x59f1a3	vfmadd23ips	Ymm1, Ymm3, Ymm5	335.339ms	5,394.1	
0x59f1a8	vfmadd23ips	Ymm2, Ymm3, Ymm12	70.833ms	1,378.1	
0x59f1ad	vbroadcasttsl	-0x4c(%rbp), Ymm3	434.864ms	5,460.1	
0x59f1b3	prefetcht02	0x280(%rbx)	20.622ms	236.1	
0x59f1ba	vfmadd23ips	Ymm0, Ymm3, Ymm5	229.640ms	3,143.1	
0x59f1b8	vfmadd23ips	Ymm1, Ymm3, Ymm9	99.526ms	1,162.1	
0x59f1c4	vfmadd23ips	Ymm2, Ymm3, Ymm13	371.030ms	4,826.1	
0x59f1c9	vbroadcasttsl	-0x48(%rbp), Ymm3	58.281ms	472.1	
0x59f1cf	vfmadd23ips	Ymm0, Ymm3, Ymm6	215.190ms	2,939.1	
0x59f1d4	vfmadd23ips	Ymm1, Ymm3, Ymm10	237.606ms	2,959.1	
0x59f1d9	vfmadd23ips	Ymm2, Ymm3, Ymm4	300.370ms	3,845.1	
0x59f1de	vbroadcasttsl	-0x44(%rbp), Ymm3	118.956ms	1,516.1	
0x59f1e4	sub	\$0xfffffffffffffc0, %rbp	144.357ms	1,904.1	
0x59f1e8	vfmadd23ips	Ymm0, Ymm3, Ymm7	156.013ms	2,189.1	
0x59f1f5	vmovupsy	0x180(%rbx), Ymm0	527.216ms	6,026.1	
0x59f1fa	vfmadd23ips	Ymm1, Ymm3, Ymm11	44.831ms	496.1	
0x59f202	vfmadd23ips	Ymm2, Ymm3, Ymm15	560.392ms	6,839.1	

## Observations and Results

Vishwanath Saikiran Shetiya

# Pipeline View for Hot Loop

## Instruction Details

Instructions	Operands	Execution Units	Port Number	Latency
VFMADD231PS	ymm1,ymm2, ymm3	FP FMA	P0, P1	4
PREFETCHT0	-	-	-	-
VBROADCASTSS	ymm,m32	LOAD	P2, P3	3
VMOVUPS	y, m256	LOAD	P2, P3	3

# Pipeline View for Hot Loop

## Pipeline view 1

Address	Assembly	Clock Cycles->																				
			cc1	cc2	cc3	cc4	cc5	cc6	cc7	cc8	cc9	cc10	cc11	cc12	cc13	cc14	cc15	cc16	cc17	cc18	cc19	cc20
0x59040	Block 1:																					
1 0x59040	vfmadd231ps ymm4, ymm3, ymm0		Issue P0	P0	P0	P0	P0	Retire														
2 0x59045	vfmadd231ps ymm8, ymm3, ymm1		Issue P1	P1	P1	P1	P1	Retire														
3 0x5904a	vfmadd231ps ymm12, ymm3, ymm2		Issue SH	P0	P0	P0	P0	Retire														
4 0x5904f	prefetch0 zmmword ptr [rbx+0x100]		Issue X	X	X	X	X	Retire														
5 0x59056	vbrodcastss ymm3, dword ptr [rbp+0x7c]		Issue P2	P2	P2	P2	X	Retire														
6 0x5905c	prefetch0 zmmword ptr [rbx+0x240]		Issue X	X	X	X	X	Retire														
7 0x59063	vfmadd231ps ymm5, ymm3, ymm0		Issue RAW	RAW	RAW	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
8 0x59068	vfmadd231ps ymm9, ymm3, ymm1		Issue RAW	RAW	RAW	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
9 0x5906d	vfmadd231ps ymm13, ymm3, ymm0		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
10 0x59072	vbrodcastss ymm3, dword ptr [rbp+0x70]		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
11 0x59078	vfmadd231ps ymm6, ymm3, ymm0		Issue RAW	RAW	RAW	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
12 0x5907d	vfmadd231ps ymm10, ymm3, ymm1		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
13 0x59082	vfmadd231ps ymm14, ymm3, ymm0		Issue RAW	RAW	RAW	SH	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
14 0x59087	vbrodcastss ymm3, dword ptr [rbp+0x74]		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
15 0x5908d	prefetch0 zmmword ptr [rbx+0x280]		Issue X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
16 0x59094	vfmadd231ps ymm7, ymm3, ymm0		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
17 0x59099	vmovups ymm0, ymmword ptr [rbx+0x20]		Issue P3	P3	P3	P3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
18 0x5909e	vfmadd231ps ymm11, ymm3, ymm1		Issue RAW	RAW	RAW	SH	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
19 0x590a3	vmovups ymm1, ymmword ptr [rbx]		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
20 0x590a7	vfmadd231ps ymm15, ymm3, ymm0		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
21 0x590ac	vmovups ymm2, ymmword ptr [rbx+0x20]		Issue P3	P3	P3	P3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
22 0x590b1	vbrodcastss ymm3, dword ptr [rbp+0x70]		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
23 0x590b7	vfmadd231ps ymm4, ymm3, ymm0		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
24 0x590bc	vfmadd231ps ymm8, ymm3, ymm1		Issue RAW	RAW	RAW	SH	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
25 0x590c1	vfmadd231ps ymm12, ymm3, ymm2		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
26 0x590c6	vbrodcastss ymm3, dword ptr [rbp+0x8c]		Issue RAW	RAW	RAW	SH	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0	P0
27 0x590cc	vfmadd231ps ymm6, ymm3, ymm0		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
28 0x590d1	vfmadd231ps ymm9, ymm3, ymm1		Issue RAW	RAW	RAW	SH	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1	P1
29 0x590d6	vfmadd231ps ymm13, ymm3, ymm2		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
30 0x590db	vbrodcastss ymm3, dword ptr [rbp+0x68]		Issue X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
31 0x590e1	prefetch0 zmmword ptr [rbx+0xc0]		Issue P2	P2	P2	P2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

# Pipeline View for Hot Loop

## Pipeline view 2

Address	Assembly		cc8	cc9	cc10	cc11	cc12	cc13	cc14	cc15	cc16	cc17	cc18	cc19	cc20	cc21	cc22	cc23	cc24	cc25	cc26
32 0x59f0e8	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	P1	P1	P1	P1	Retire										
33 0x59f0ed	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire										
34 0x59f0f2	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire										
35 0x59f0f7	vbroadcastss ymm3, dword ptr [rbp-0x64]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
36 0x59f0fd	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire									
37 0x59f102	vmovups ymm0, ymmword ptr [rbx+0x40]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
38 0x59f107	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire										
39 0x59f10c	vmovups ymm1, ymmword ptr [rbx+0x60]	Issue	P3	P3	P3	X	X	X	X	X	Retire										
40 0x59f111	vfmadd231ps ymm15, ymm3, ymm2	Issue	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire										
41 0x59f116	vmovups ymm2, ymmword ptr [rbx+0x80]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
42 0x59f11e	vbroadcastss ymm3, dword ptr [rbp-0x60]	Issue	P3	P3	P3	X	X	X	X	X	Retire										
43 0x59f124	vfmadd231ps ymm4, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	P1	P1	P1	P1	Retire										
44 0x59f129	vfmadd231ps ymm8, ymm3, ymm1	Issue	RAW	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire									
45 0x59f12e	prefetcht0 zmmword ptr [rbx+0x300]	Issue	X	X	X	X	X	X	X	X	Retire										
46 0x59f135	vfmadd231ps ymm12, ymm3, ymm2	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire										
47 0x59f13a	vbroadcastss ymm5, dword ptr [rbp-0x5c]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
48 0x59f140	vfmadd231ps ymm5, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire									
49 0x59f145	vfmadd231ps ymm9, ymm3, ymm1	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire										
50 0x59f14a	vfmadd231ps ymm13, ymm3, ymm2	Issue	RAW	RAW	SH	SH	P0	P0	P0	P0	Retire										
51 0x59f14f	vbroadcastss ymm3, dword ptr [rbp-0x58]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
52 0x59f155	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire									
53 0x59f15a	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	SH	SH	P0	P0	P0	P0	P0	Retire									
54 0x59f15f	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	P1	Retire									
55 0x59f164	vbroadcastss ymm3, dword ptr [rbp-0x54]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
56 0x59f16a	prefetcht0 zmmword ptr [rbx+0x340]	Issue	X	X	X	X	X	X	X	X	Retire										
57 0x59f171	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire									
58 0x59f176	vmovups ymm0, ymmword ptr [rbx+0xa0]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
59 0x59f17e	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	SH	SH	P1	P1	P1	P1	P1	Retire									
60 0x59f183	vmovups ymm1, ymmword ptr [rbx+0xc0]	Issue	P3	P3	P3	X	X	X	X	X	Retire										
61 0x59f18b	vfmadd231ps ymm15, ymm3, ymm2	Issue	RAW	SH	SH	SH	SH	P0	P0	P0	P0	Retire									
62 0x59f190	vmovups ymm2, ymmword ptr [rbx+0xe0]	Issue	P2	P2	P2	X	X	X	X	X	Retire										
63 0x59f198	vbroadcastss ymm3, dword ptr [rbp-0x50]	Issue	P3	P3	P3	X	X	X	X	X	Retire										

# Pipeline View for Hot Loop

## Pipeline view 3

Address	Assembly	Pipeline Stages (cc13 to cc33)																				
		cc13	cc14	cc15	cc16	cc17	cc18	cc19	cc20	cc21	cc22	cc23	cc24	cc25	cc26	cc27	cc28	cc29	cc30	cc31	cc32	cc33
64 0x59f19e	vfmadd231ps ymm4, ymm3, ymm0	Issue	RAW	RAW	RAW	SH	SH	P1	P1	P1	P1	Retire										
65 0x59f1a3	vfmadd231ps ymm8, ymm3, ymm1	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire										
66 0x59f1a8	vfmadd231ps ymm12, ymm3, ymm2	Issue	RAW	RAW	SH	SH	SH	P1	P1	P1	P1	Retire										
67 0x59f1ad	vbroadcastss ymm3, dword ptr [rbp-0x4c]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
68 0x59f1b3	prefetcht0 zmmword ptr [rbx+0x380]	Issue	X	X	X	X	X	X	X	X	X	X	Retire									
69 0x59f1ba	vfmadd231ps ymm5, ymm3, ymm0	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire										
70 0x59f1bf	vfmadd231ps ymm9, ymm3, ymm1	Issue	RAW	RAW	SH	SH	SH	P1	P1	P1	P1	Retire										
71 0x59f1c4	vfmadd231ps ymm13, ymm3, ymm2	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire										
72 0x59f1c9	vbroadcastss ymm3, dword ptr [rbp-0x48]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
73 0x59f1cf	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	SH	SH	SH	P1	P1	P1	P1	Retire										
74 0x59f1d4	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire										
75 0x59f1d9	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	SH	SH	SH	P1	P1	P1	P1	Retire										
76 0x59f1de	vbroadcastss ymm3, dword ptr [rbp-0x44]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
77 0x59f1e4	sub rbp, 0xfffffffffc0	Issue	P5	X	X	X	X	X	X	X	X	X	Retire									
78 0x59f1e8	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	SH	SH	SH	P0	P0	P0	P0	Retire										
79 0x59f1ed	vmovups ymm0, ymmword ptr [rbx+0x100]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
80 0x59f1f5	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	SH	SH	SH	P1	P1	P1	P1	Retire										
81 0x59f1fa	vmovups ymm1, ymmword ptr [rbx+0x120]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
82 0x59f202	vfmadd231ps ymm15, ymm3, ymm2	Issue	RAW	SH	SH	SH	SH	SH	P0	P0	P0	P0	Retire									
83 0x59f207	vmovups ymm2, ymmword ptr [rbx+0x140]	Issue	P3	P3	P3	X	X	X	X	X	X	X	Retire									
84 0x59f20f	sub rbx, 0xfffffffffe0	Issue	P5	X	X	X	X	X	X	X	X	X	Retire									
85 0x59f216	vbroadcastss ymm3, dword ptr [rbp-0x80]	Issue	P2	P2	P2	X	X	X	X	X	X	X	Retire									
86 0x59f21c	sub rax, 0x1	Issue	P5	X	X	X	X	X	X	X	X	X	Retire									
87 0x59f220	hint-taken jne 0x59f040 <Block 11>	Issue	P6	X	X	X	X	X	X	X	X	X	Retire									

# Summary of Pipeline View for Hot Loop

## Too Many Structural Hazards

Total number of RAW Stalls	104
Total number of Strucutral hazard(SH) stalls	98
Total Number of Stall X	231
Ratio of Raw stall	0.2401847575
% of RAW stalls	24.01847575
Ratio of SH stalls	0.2263279446
% of SH stalls	22.63279446
IPC(No of Instruction / No of Clock cycle)	2.636363636

# Proposed Solution

## Pipeline View 1 - adding FP FMA at Port. No 8

Address	Assembly	Clock Cycles->														
		cc1	cc2	cc3	cc4	cc5	cc6	cc7	cc8	cc9	cc10	cc11	cc12	cc13	cc14	cc15
0x59f040	<b>Block 11:</b>															
1 0x59f040	vfmadd231ps ymm4, ymm3, ymm0															
2 0x59f045	vfmadd231ps ymm8, ymm3, ymm1															
3 0x59f04a	vfmadd231ps ymm12, ymm3, ymm2															
4 0x59f04d	prefetcht0 zmmword ptr [rbp+0x100]															
5 0x59f056	vbroadcastss ymm3, dword ptr [rbp+0x7c]															
6 0x59f05c	prefetcht0 zmmword ptr [rbx+0x240]															
7 0x59f063	vfmadd231ps ymm5, ymm3, ymm0															
8 0x59f068	vfmadd231ps ymm9, ymm3, ymm1															
9 0x59f06d	vfmadd231ps ymm13, ymm3, ymm2															
10 0x59f072	vbroadcastss ymm3, dword ptr [rbp+0x78]															
11 0x59f078	vfmadd231ps ymm6, ymm3, ymm0															
12 0x59f07d	vfmadd231ps ymm10, ymm3, ymm1															
13 0x59f082	vfmadd231ps ymm14, ymm3, ymm2															
14 0x59f087	vbroadcastss ymm3, dword ptr [rbp+0x74]															
15 0x59f08d	prefetcht0 zmmword ptr [rbx+0x280]															
16 0x59f094	vfmadd231ps ymm7, ymm3, ymm0															
17 0x59f099	vmovups ymm0, ymmword ptr [rbx+0x20]															
18 0x59f09e	vfmadd231ps ymm11, ymm3, ymm1															
19 0x59f0a3	vmovups ymm1, ymmword ptr [rbx]															
20 0x59f0a7	vfmadd231ps ymm15, ymm3, ymm2															
21 0x59f0a8	vmovups ymm2, ymmword ptr [rbx+0x20]															
22 0x59f0b1	vbroadcastss ymm3, dword ptr [rbp+0x70]															
23 0x59f0b7	vfmadd231ps ymm4, ymm3, ymm0															
24 0x59f0b8	vfmadd231ps ymm8, ymm3, ymm1															
25 0x59f0c1	vfmadd231ps ymm12, ymm3, ymm2															
26 0x59f0c6	vbroadcastss ymm3, dword ptr [rbp+0x6c]															
27 0x59f0c8	vfmadd231ps ymm5, ymm3, ymm0															
28 0x59f0d1	vfmadd231ps ymm9, ymm3, ymm1															
29 0x59f0d6	vfmadd231ps ymm13, ymm3, ymm2															
30 0x59f0db	vbroadcastss ymm3, dword ptr [rbp+0x68]															
31 0x59f0e1	prefetcht0 zmmword ptr [rbx+0xc0]															

# Proposed Solution

## Pipeline View 2 - adding FP FMA at Port. No 8

Address	Assembly	cc8	cc9	cc10	cc11	cc12	cc13	cc14	cc15	cc16	cc17	cc18	cc19	cc20	cc21	cc22	cc23	cc24
32 0x59f0e8	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	Retire								
33 0x59f0ed	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	Retire									
34 0x59f0f2	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	Retire									
35 0x59f0f7	vbroadcastss ymm3, dword ptr [rbp-0x64]	Issue	P2	P2	P2	X	X	X	X	Retire								
36 0x59f0fd	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	Retire								
37 0x59f102	vmovups ymm0, ymmword ptr [rbx+0x40]	Issue	P2	P2	P2	X	X	X	X	Retire								
38 0x59f107	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	Retire									
39 0x59f10c	vmovups ymm1, ymmword ptr [rbx+0x60]	Issue	P3	P3	P3	X	X	X	X	Retire								
40 0x59f111	vfmadd231ps ymm15, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	X	Retire								
41 0x59f116	vmovups ymm3, ymmword ptr [rbx+0x80]	Issue	P2	P2	P2	X	X	X	X	Retire								
42 0x59f11e	vbroadcastss ymm3, dword ptr [rbp-0x60]	Issue	P3	P3	P3	X	X	X	X	Retire								
43 0x59f124	vfmadd231ps ymm4, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	Retire								
44 0x59f129	vfmadd231ps ymm8, ymm3, ymm1	Issue	RAW	RAW	RAW	P1	P1	P1	P1	Retire								
45 0x59f12e	prefetcht0 zmmword ptr [rbx+0x300]	Issue	X	X	X	X	X	X	X	Retire								
46 0x59f135	vfmadd231ps ymm12, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	Retire								
47 0x59f13a	vbroadcastss ymm3, dword ptr [rbp-0x5c]	Issue	P2	P2	P2	X	X	X	X	Retire								
48 0x59f140	vfmadd231ps ymm5, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	Retire								
49 0x59f145	vfmadd231ps ymm9, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	Retire								
50 0x59f14a	vfmadd231ps ymm13, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	Retire								
51 0x59f14f	vbroadcastss ymm3, dword ptr [rbp-0x58]	Issue	P2	P2	P2	X	X	X	X	Retire								
52 0x59f155	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	Retire								
53 0x59f15a	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	Retire								
54 0x59f15f	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	Retire								
55 0x59f164	vbroadcastss ymm3, dword ptr [rbp-0x54]	Issue	P2	P2	P2	X	X	X	X	Retire								
56 0x59f16a	prefetcht0 zmmword ptr [rbx+0x340]	Issue	X	X	X	X	X	X	X	Retire								
57 0x59f171	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	P0	P0	P0	P0	P0	Retire								
58 0x59f176	vmovups ymm0, ymmword ptr [rbx+0xa0]	Issue	P2	P2	P2	X	X	X	X	Retire								
59 0x59f17e	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	Retire	X							
60 0x59f183	vmovups ymm1, ymmword ptr [rbx+0xc0]	Issue	P3	P3	P3	X	X	X	X	Retire	X							
61 0x59f18b	vfmadd231ps ymm15, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	X	Retire							
62 0x59f190	vmovups ymm2, ymmword ptr [rbx+0xe0]	Issue	P2	P2	P2	X	X	X	X	Retire	X							

# Proposed Solution

## Pipeline View 3 - adding FP FMA at Port. No 8

Address	Assembly	-15	cc16	cc17	cc18	cc19	cc20	cc21	cc22	cc23	cc24	cc25	cc26	cc27	cc28	cc29	cc30
64 0x59f19e	vfmadd231ps ymm4, ymm3, ymm0	Issue	RAW	RAW	RAW	P0	P0	P0	P0	P0	Retire						
65 0x59f1a3	vfmadd231ps ymm8, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	P1	Retire						
66 0x59f1a8	vfmadd231ps ymm12, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	P8	Retire						
67 0x59f1ad	vbroadcastss ymm3, dword ptr [rbp-0x4c]	Issue	P2	P2	P2	X	X	X	X	X	Retire						
68 0x59f1b3	prefetch0 zmmword ptr [rbx+0x380]	Issue	X	X	X	X	X	X	X	X	Retire						
69 0x59f1ba	vfmadd231ps ymm5, ymm3, ymm0	Issue	RAW	RAW	P0	P0	P0	P0	P0	P0	Retire						
70 0x59f1bf	vfmadd231ps ymm9, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	P1	Retire						
71 0x59f1c4	vfmadd231ps ymm13, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	P8	X	Retire					
72 0x59f1c9	vbroadcastss ymm3, dword ptr [rbp-0x48]	Issue	P2	P2	P2	X	X	X	X	X	Retire						
73 0x59f1cf	vfmadd231ps ymm6, ymm3, ymm0	Issue	RAW	RAW	P0	P0	P0	P0	P0	P0	Retire						
74 0x59f1d4	vfmadd231ps ymm10, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	P1	Retire						
75 0x59f1d9	vfmadd231ps ymm14, ymm3, ymm2	Issue	RAW	RAW	P8	P8	P8	P8	P8	P8	X	Retire					
76 0x59f1de	vbroadcastss ymm3, dword ptr [rbp-0x44]	Issue	P2	P2	P2	X	X	X	X	X	Retire						
77 0x59f1e4	sub rbp, 0xfffffffffffffc0	Issue	P5	X	X	X	X	X	X	X	Retire						
78 0x59f1e8	vfmadd231ps ymm7, ymm3, ymm0	Issue	RAW	RAW	P0	P0	P0	P0	P0	P0	Retire						
79 0x59f1ed	vmovups ymm0, ymmword ptr [rbx+0x100]	Issue	P2	P2	P2	X	X	X	X	X	Retire						
80 0x59f1f5	vfmadd231ps ymm11, ymm3, ymm1	Issue	RAW	RAW	P1	P1	P1	P1	P1	P1	X	Retire					
81 0x59f1fa	vmovups ymm1, ymmword ptr [rbx+0x120]	Issue	P2	P2	P2	X	X	X	X	X	Retire						
82 0x59f202	vfmadd231ps ymm5, ymm3, ymm2	Issue	RAW	P8	X	Retire											
83 0x59f207	vmovups ymm2, ymmword ptr [rbx+0x140]	Issue	P3	P3	P3	X	X	X	X	X	X	Retire					
84 0x59f20f	sub rbx, 0xfffffffffffffe80	Issue	P5	X	X	X	X	X	X	X	X	Retire					
85 0x59f216	vbroadcastss ymm3, dword ptr [rbp-0x80]	Issue	P2	P2	P2	X	X	X	X	X	X	Retire					
86 0x59f21c	sub rax, 0x1	Issue	P5	X	X	X	X	X	X	X	X	Retire					
87 0x59f220	hint-taken jne 0x59f040 <Block 11>	Issue	P6	X	X	X	X	X	X	X	X	Retire					

## Summary of Pipeline View for Hot Loop after new addition

We have got an IPC gain of 10%. The Total Number of clock cycles is 30. For the hotspot function, we got an IPC gain of 6.59%

Total number of RAW Stalls	104
Total number of Strucutral hazard(SH) stalls	0
Total Number of Stall X	167
Ratio of Raw stall	0.3837638376
% of RAW stalls	38.37638376
Ratio of SH stalls	0
% of SH stalls	0
IPC(No of Instruction / No of Clock cycle)	2.9
IPC gain	0.1000000002
Hotspot %CPUTime	0.659
Application IPC gain	0.0659
IPC gain in %	10.00000002
Application IPC gain %	6.59

## 1 Background and Motivation

## 2 Literature Survey

### 3 Methodology and Profilers

4 Preliminary Setup

## 5 Observations and Results

## 6 Conclusions and Future Works

## Conclusions

- Insufficient port utilization is the primary bottleneck for the BERT base uncased model.
  - The Introduction of extra units can give us a good improvement in the inference task.

## Future Works

- Since multithreading was disabled for our experiments, we shall enable it and repeat the same microarchitecture analysis and confirm if the results that we got holds.
  - Use a trace-based simulator like Champsim, DynamoRio, or GEM5 to confirm that the improvement persists.

## Acknowledgement

## My Guide:

- ① Dr. R. Raghunatha Sarma, Associate Professor, Mathematics and Computer Science, SSSIHL.

## Mentors:

- ① Mr. Dibyam Pradhan, Principal CPU Architect, ARM.
  - ② Mr. Naveen M, Senior Member of Technical Staff, AMD.
  - ③ Mr. Mangala Prasad Sahu, Hardware Developer IBM.
  - ④ Mr. Aravind S. V, Graduate Engineer, ARM.

## References

- Ashish Vaswani, Noam Shazeer, Niki Parmar, Llion Jones, Aidan N. Gomez , ukasz Kaiser , Illia Polosukhin. 2017. Attention is all you need. NIPS.
  - A. Yasin, A top-down method for performance analysis and counters architecture, in 2014 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS).
  - J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, Bert: Pre- training of deep bidirectional transformers for language understanding, arXiv preprint, vol. arXiv:1810.04805, 2018. [Online].
  - J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, 5th ed. Morgan Kaufmann, 2012.
  - A. Fog, Instruction Tables: Lists of instruction latencies, throughputs and micro-operation breakdowns for Intel, AMD and VIA CPUs, 2024.
  - Intel Corporation, Developer documentation, <https://www.intel.com/content/www/us/en/resources-documentation/developer.html>, 2024, accessed: 2025-04-10.

## References

- Wikipedia contributors, Skylake (microarchitecture) wikipedia, the free encyclopedia, 2024, accessed: 2025-04-10. [Online].
  - I. Corporation, Element-wise primitive (eltwise), 2023, accessed: 2025- 04-09. [Online]. Available: <https://www.intel.com/content/www/us/en/docs/onednn/developer-guide-reference/2023-2/eltwise-001.html>
  - I. Corporation, Top-down microarchitecture analysis method, <https://www.intel.com/content/www/us/en/docs/vtune-profiler/cookbook/2023-0/top-down-microarchitecture-analysis-method.html>, 2023.
  - J. Alammar, The illustrated transformer, <https://jalammar.github.io/illustrated-transformer/>, 2018.

Thank you!