Google TPU

Fernando Lima
Isabella Caselli
Rodrigo Michelassi

2024

Conteúdo

1 Introdução **1**

2 História **2**
2.1 Tensores 2
2.2 Aprendizado de Máquina e Redes Neurais 2

3 Arquitetura da TPU **4**
3.1 Unidade de Multiplicação Matricial 4
3.2 Principais instruções 5
3.3 Compatibilidade 6

4 TPU vs GPU **7**
4.1 Arquitetura e Design 7
4.2 Desempenho 7
4.3 Custos 7

5 TensorFlow **9**

6 Cloud TPU v5p **10**

7 Consumo de Energia **11**

8 Considerações Finais **13**

Resumo

Na era do desenvolvimento de sistemas baseados em Inteligência Artificial e redes neurais profundas (Deep Learning), se faz necessário o uso de máquinas super potentes, capazes de processar dados e realizar operações matemáticas de maneira extremamente rápida. Modelos de Machine Learning podem levar horas, até mesmo dias, para serem treinados, devido principalmente a operações como produto interno entre matrizes e a enorme quantidade de dados que são usados, trazendo um prejuízo não apenas de tempo, mas também energético, ambiental e sobretudo lucrativo. Nesse artigo, iremos tratar brevemente sobre a utilização de Cloud TPUs, unidades de processamento de tensores do Google Cloud, que atuam na otimização do treinamento de modelos de aprendizado de máquina, e que se tornou indispensável na academia e na indústria, para todos estudiosos e profissionais da área.

Capítulo 1

Introdução

A evolução da tecnologia tem proporcionado diversos avanços no campo da inteligência artificial, principalmente no desenvolvimento de redes neurais profundas (Deep Learning) e aprendizado de máquina. Nesse cenário, a demanda por maior poder computacional levou à criação de hardware especializado, capaz de lidar com a complexidade e o volume de cálculos necessários para essas tarefas. Nesse contexto, destaca-se a Unidade de Processamento Tensorial (TPU), um circuito integrado de aplicação específica (ASIC) desenvolvido pelo Google.

As TPUs, lançadas em 2015, foram projetadas para acelerar operações de aprendizado de máquina, otimizando o treinamento de modelos baseados em redes neurais. Diferentemente de CPUs e GPUs, cuja arquitetura é projetada para fins generalistas, as TPUs são especializadas em multiplicações e somas de matrizes, que são operações fundamentais em áreas como Deep Learning. Essa especialização confere às TPUs maior eficiência energética e desempenho significativamente superior em tarefas relacionadas à inteligência artificial.

Dessa forma, considerando o tema apresentado como essencial para os avanços significativos que foram alcançados nas áreas de IA e Aprendizado de Máquina, este trabalho se propõe a explorar a arquitetura, os princípios de funcionamento e as aplicações das TPUs no contexto da computação moderna.

Capítulo 2

História

A Google Tensor Processing Unit é uma arquitetura de circuito integrado conhecido como um "acelerador de IA". Essa arquitetura tem seu início em 2016, como uma alternativa a outras estruturas já conhecidas, utilizadas popularmente para acelerar o treinamento de modelos de IA, como as GPUs e os arrays sistólicos.

Apesar dessa tecnologia ser divulgada ao público apenas em 2016, engenheiros da Google divulgaram que já era utilizada há mais de um ano em datacenters da big tech. Essa arquitetura foi pensada para funcionar juntamente a biblioteca de Machine Learning da Google, o TensorFlow, utilizada para treinar grandes modelos baseados em redes neurais, e atualmente uma das maiores bibliotecas do ramo.

Entre as principais atividades executadas pela TPU está o processamento de tensores, estrutura de dados conhecida como um array multidimensional. Dessa forma, estratégias diversas para multiplicação de tensores (linear e paralelamente) são comumente estudadas em diversos campos da matemática clássica, dentro da Álgebra Linear, e as mais modernas arquiteturas para processamentos de produtos matriciais se baseiam em técnicas já conhecidas.

Atualmente, as TPUs são proprietárias, e no geral o acesso se limita a própria Google ou usuários que pagam por seu uso. É possível que empresas contratem o serviço por meio do sistema de nuvem da Google ou utilizem através do Google Colab, com tempo limitado de uso ou aluguel de TPUs mais potentes.

2.1 Tensores

2.2 Aprendizado de Máquina e Redes Neurais

Nos tempos de ChatGPT, IA's generativas, problemas éticos com uso de imagens e dados pessoais por grandes empresas, surge a necessidade de se entender como funcionam os modelos de Aprendizado de Máquina.

Nesse sentido, o surgimento das TPUs estão associadas justamente oa fortalecimento científico e comercial desse tipo de tecnologia, que a cada dia vem sendo explorada mais fortemente, e utilizando grandes camadas de dados. Esses modelos exigem muito poder computacional para processar todos os dados fornecidos, além de realizar diversas operações matemáticas como produtos tensoriais.

De maneira resumida, modelos de Aprendizado de Máquina no geral consiste na modelagem matemática de problemas, com propósitos variáveis, mas que todos possuem algo em comum: fazer com que o modelo se ajuste bem, a ponto de dar respostas boas para suas entradas. Dessa forma, se percebe uma boa semelhança com a computação clássica, dada uma entrada, processamos uma respectiva saída. Todavia, esses modelos são probabilísticos, e não há garantia que a resposta sempre será a esperada. Para que servem esses modelos, então?

Há problemas que são fáceis para humanos responderem, mas difíceis de serem automatizados. Por exemplo, dada uma imagem de um gato, um ser humano pode facilmente ver a imagem e dizer "essa é uma imagem de um gato". Porém, se existisse a necessidade de automatizar tal tarefa? Como podemos escrever um algoritmo que, dada uma imagem, de qualquer tamanho e padrão de cores, apenas utilizando ferramentas clássicas de algoritmos (loops, condicionais) e dizer se aquela imagem contém de fato, um gato? Nesse contexto, utilizar Machine Learning é uma boa ideia.

Adicionalmente, os modelos mais modernos de Aprendizado de Máquina estão atrelados ao desenvolvimento de redes neurais. Esses modelos são treinados baseados em dois passos, conhecidos como forward e back-propagation. Redes Neurais são constituídas por nós, representando features de dados, e pesos atribuídos a cada uma dessas features, e a operação de Forward-Propagation consiste em realizar produtos vetoriais entre esses pesos com o valor dos nós (função não linear da soma ponderada dos dados de entrada). A operação de Back-Propagation deve retornar a rede neural, atualizando os pesos com base no erro calculado, novamente utilizando operações de produtos vetoriais. Note que essas operações são extremamente custosas, e Redes Neurais possuem milhares, até milhões de parâmetros, a depender do modelo, e é necessário acelerar esse processamento, com base em paralelismo e busca de algoritmos mais eficientes.

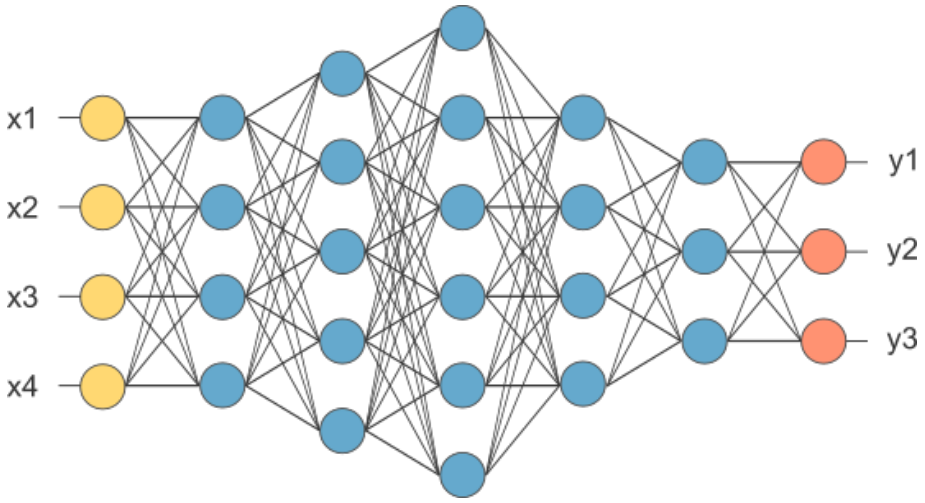

Figura 2.1: Estrutura gráfica de uma Rede Neural

Com base nisso, temos um modelo matemático, que atualiza pesos e é capaz de responder perguntas, porém se ele nem sempre acerta, como eles são de fato utilizados? Vimos que modelos de Aprendizado de Máquina são baseados matemáticos e baseados em pesos. Existem diversas aplicações para esses modelos, e o trabalho de um engenheiro de Machine Learning é conseguir treinar um modelo, com dados o suficiente, de forma a minimizar o erro desse modelo, dessa forma aumentando sua acurácia e garantindo que o modelo tenha uma maior probabilidade de acertar as respostas, com base nos dados que são utilizados como entrada. Atualmente, esses modelos tem diversos usos na indústria, como no mercado financeiro, para predição de séries temporais para variações da bolsa de valores, na medicina para auxílio de profissionais no diagnóstico de doenças, no direito para análise textual de casos, na astronomia, como na recente descoberta da primeira imagem de um buraco negro.

Dessa forma, Machine Learning é uma tendência que cresce tanto no mercado como na academia, e deve continuar sendo explorado por diversos profissionais, de forma que é necessário acelerar o longo processo de treinamento de modelos, além de buscar soluções mais sustentáveis para que o conhecimento possa continuar se expandindo.

Capítulo 3

Arquitetura da TPU

As TPUs surgem como uma alternativa de arquitetura simples para o usuário, de forma que tenha uma interface de hardware mais amigável, ao mesmo tempo que consegue processar dados mais rapidamente, atingindo as demandas necessárias em 2015 para processamento de redes neurais. Essa arquitetura foi desenhada para processar dados de maneira independente da CPU, e recebe instruções de processamento do servidor onde está conectado.

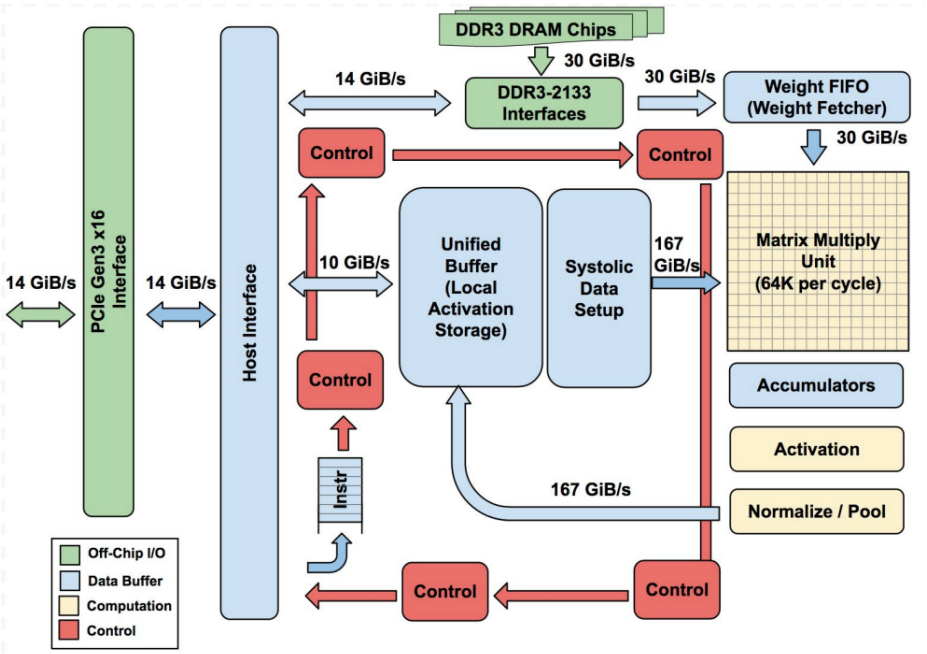

Figura 3.1: Estrutura básica de uma TPU

Para entender melhor a arquitetura de uma TPU, leve em consideração seus pontos mais importantes: a entrada é feita no Weight FIFO, que é levada a principal unidade de processamento, a Matrix Multiply Unit. Essa unidade é responsável por explorar o produto matricial, e iremos explicar mais a fundo em breve. Por fim, a saída é deixada nos acumuladores, e a unidade de ativação performa funções não lineares na saída, para finalmente levar essa saída ao Buffer unificado.

3.1 Unidade de Multiplicação Matricial

Como dissemos, a unidade de multiplicação matricial é o coração da TPU. Essa unidade contém 256×256 MACs (endereço de controle de acesso de mídia) que conseguem processar operações de adição de multiplicação em inteiros (com ou sem sinal) de até 8 bits, gerando produtos de até 16 bits,

armazenados temporariamente nos acumuladores de 32 bits. Esses acumuladores podem carregar até 4MiB de dados. A unidade de processamento de matrizes pode computar até 256 valores por ciclo de clock, e é capaz de realizar operações como produto matricial ou convolução. Vale lembrar que, nos primeiros anos do lançamento da TPU, essa unidade foi planejada justamente para acelerar operações de produto em matrizes densas, sem levar em consideração a esparsidade, que é uma medida útil para acelerar operações de matrizes.

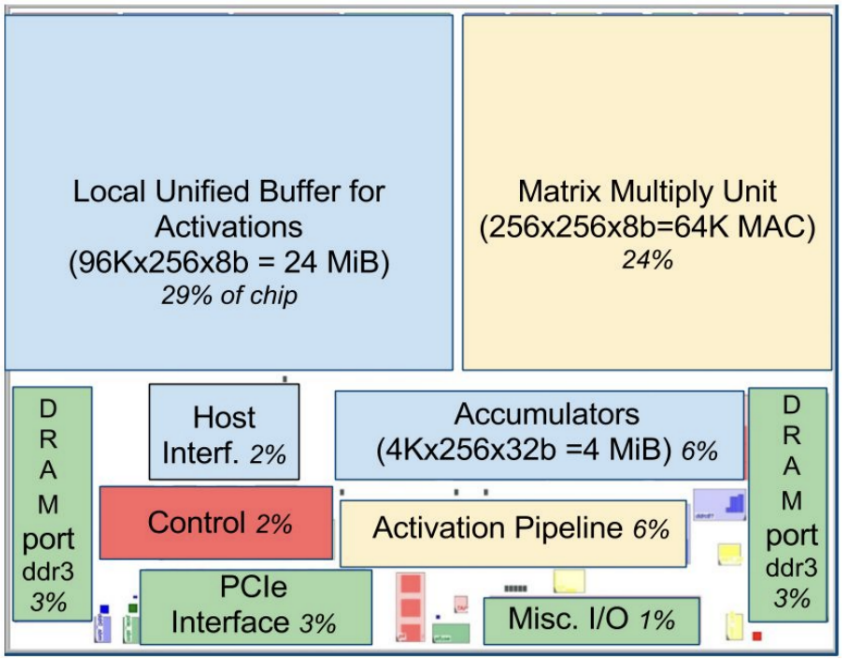

Figura 3.2: Divisão do uso do chip na TPU por unidade

Note que, a unidade de processamento de matrizes irá receber muitos dados de entrada, e ler dados da SRAM consome muito mais energia do que processamento aritmético, portanto essa unidade se aproveita de execução sistólica para economizar energia, reduzindo operações de leitura e escrita do Buffer unificado, dependendo de dados provindos de diversas direções alimentando um array em intervalos regulares.

3.2 Principais instruções

Como vimos, as instruções a serem executadas são enviadas de maneira externa para a TPU, através de PCIe (Peripheral Component Interconnect Express), o que pode ser relativamente lento e caminhar na direção oposta ao que é esperado. Nesse contexto, a fim de acelerar esse processo, a TPU utiliza da arquitetura CISC para definir instruções. A seguir, vemos algumas das principais instruções presentes na TPU:

- Read_Host_Memory: lê dados da CPU para o Buffer Unificado da TPU
- Read_Weights: lê a entrada da unidade matricial
- MatrixMultiply/Convolve: faz com que a unidade matricial performe uma operação de multiplicação ou convolução, e deposite a saída nos acumuladores. Tal operação matricial leva tempo $B \times 256$ da entrada, multiplica por uma constante 256×256 e produz uma saída de tamanho $B \times 256$, e leva B ciclos de clock para ser concluída.
- Activate: performa a função não linear de ativação no neurônio processado que está no acumulador, podendo ser, nativamente, ReLU, Sigmoid, até mesmo pooling para convoluções. Sua saída é levada para o Buffer unificado.
- Write_Host_Memory: escreve os dados do Buffer unificado na CPU.

Como podemos ver, essas principais instruções giram em torno do uso da unidade de processamento de matriz, além de trazer a aplicação de funções conhecidas de Machine Learning, como pooling

e funções de ativação, para serem executadas a nível de hardware, acelerando ainda mais o processo. A filosofia geral da TPU é manter a unidade de matrizes sempre ocupada, e processar diversas instruções paralelamente, para evitar que a unidade de matrizes não tenha entrada assim que acabar uma operação.

3.3 Compatibilidade

Note que a TPU está sendo utilizada sempre de maneira conjunta com a CPU, para recebimento de dados a serem processados. Dessa forma, a pilha de software da TPU tinha como necessidade padrão ser compatível com softwares feitos para rodar na CPU e GPU. Para evitar problemas de compatibilidade, a aplicação de roda na TPU é feita utilizando TensorFlow e compilada em uma API que é capaz de rodar na CPU e GPU.

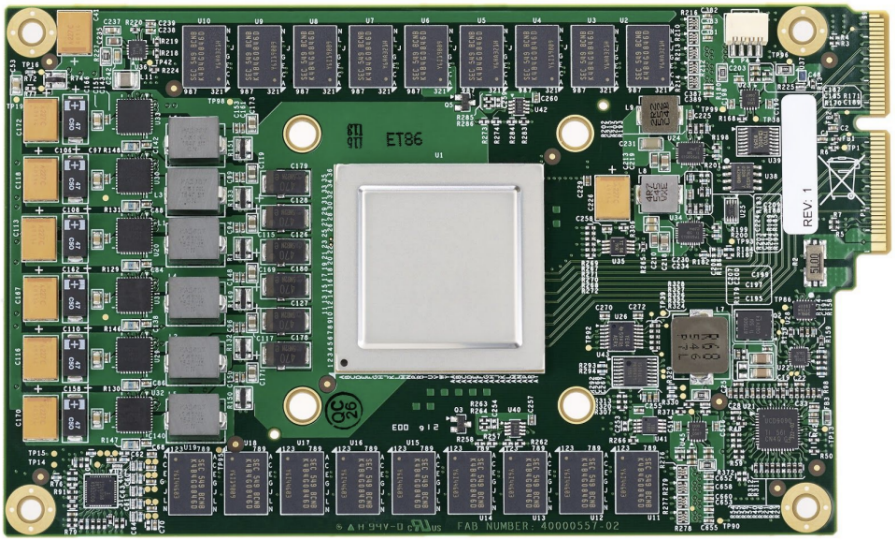

Figura 3.3: Placa de circuito da TPU

Capítulo 4

TPU vs GPU

As GPUs (Graphics Processing Unit ou Unidade de Processamento Gráfico) são processadores projetados para acelerarem o processamento e a renderização de gráficos e imagens em um computador. Inicialmente, elas foram desenvolvidas para melhorar o desempenho gráfico e aplicações de visualização, mas com o tempo, elas se tornaram ferramentas poderosas também para tarefas de computação paralela. Devido a esta grande vantagem, computar paralelamente, elas tornaram-se vitais nas tarefas de aprendizado de máquina e simulações científicas.

As TPUs e GPUs, cada uma possui vantagens distintas e são otimizadas para diferentes propósitos de computação. Embora ambas possuam alguns propósitos em comum, suas arquiteturas e otimizações levam a variações no desempenho, custo e eficiência dependendo da tarefa específica.

4.1 Arquitetura e Design

a GPU é uma unidade de processamento paralela com milhares de núcleos que podem executar tarefas simultaneamente. Sua arquitetura e design são altamente otimizados para executarem cálculos gráficos. Ela possui uma capacidade de lidar com enormes quantidades de dados e, devido à sua característica de paralelismo, torna-se adequada para operações matemáticas complexas.

As TPUs não possuem tantos núcleos quanto as GPUs, porém, devido à sua arquitetura especializada para o processamento de tensores (operações matriciais), isto permite que elas superem as GPUs em determinadas tarefas. Aprendizagem profunda (Deep Learning), redes neurais, operações de convolução e outras técnicas de aprendizagem de máquina são os principais focos das TPUs.

4.2 Desempenho

As comparações entre TPUs e GPUs em tarefas semelhantes frequentemente revelam que as TPUs superam as GPUs principalmente em problemas nos quais a TPU foi especialmente arquitetada para resolver. Logo, as TPUs superam as GPUs em tarefas específicas de aprendizagem profunda, como treinamento extensivo de redes neurais e modelos complexos de machine learning.

Um exemplo seria o treinamento de um modelo ResNet-50 no conjunto de dados CIFAR-10 utilizando uma GPU NVIDIA Tesla V100 leva aproximadamente 40 minutos. Enquanto uma TPU v3 do Google Cloud, o mesmo treinamento leva apenas 15 minutos.

4.3 Custos

As GPUs possuem custos relativamente mais baratos do que as TPUs. Um dos principais empecilhos das TPUs é que elas não são vendidas individualmente, mas são disponíveis através de serviços de Cloud. Enquanto as GPUs são vendidas individualmente e atingem valores de dezenas de milhares de dólares.

Ambas também podem ser obtidas por serviços de nuvem e a diferença no preço é chocante. Uma NVIDIA A100 pode custar cerca de 3 dólares a hora. Por outro lado, o Google Cloud TPU V4 custaria aproximadamente 8,00 por hora.

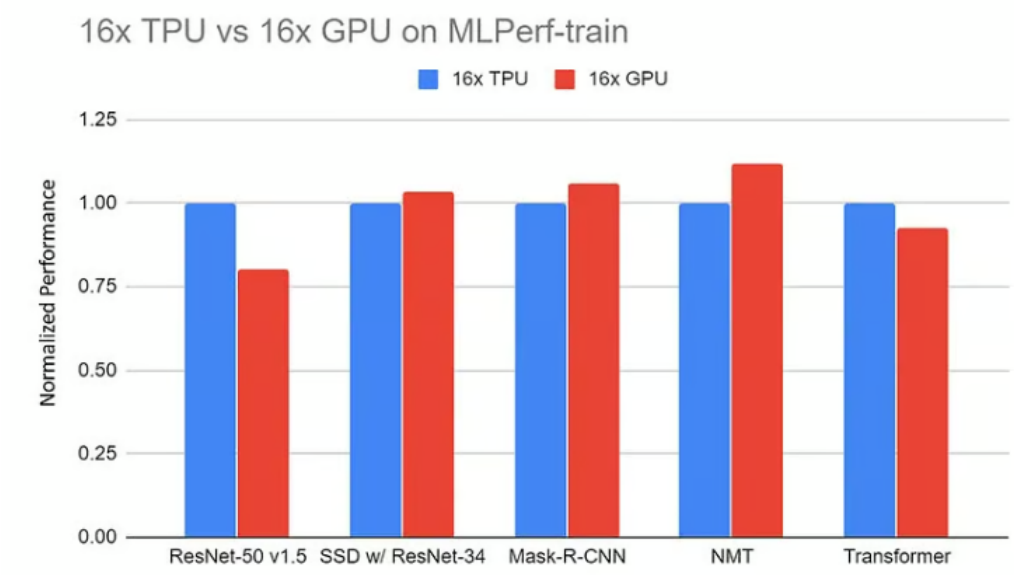

Figura 4.1: Comparação de desempenhos entre TPU e GPU

Esta diferença desproporcional está atrelada à grande eficiência das TPUs para tarefas de machine learning em grande escala. Apesar das TPUs terem o custo por hora mais caro, tais tarefas demandam muitas horas de cálculos computacionais. Como existe uma vantagem maior no desempenho das TPUs em relação as GPUs nessas tarefas, o saldo final apresenta-se positivo para a utilização das mesmas.

Capítulo 5

TensorFlow

Capítulo 6

Cloud TPU v5p

Como discutido anteriormente, LLMs são os novos modelos mais explorados no mercado e na academia. Todavia, esses modelos utilizam dados de toda (ou quase toda) a internet, que cresce ainda mais exponencialmente. Todos os dias há milhões de novos dados sendo gerados. Dessa forma, é importante que o hardware acompanhe o crescimento na quantidade de dados disponíveis, de forma que seja possível possuir poder computacional suficiente para treinar modelos como esses.

Com isso em vista, a evolução das TPUs deve ser rápida, portanto, em 2023, a Google apresentou, em 2023, a v5p, a TPU mais poderosa da empresa. Com esse lançamento, a Google prometeu a entrega de uma performance até 2.8 vezes mais rápida, utilizada para alimentar ecossistemas internos da multinacional, como o Youtube, Android e Gmail.

Essa TPU se diferencia das demais, pois é focada em desempenho possível, sem levar em consideração a facilidade operacional. Um grande diferencial é a forma como a TPU lida com operações em ponto flutuante. Por dados oficiais da Google, essa arquitetura traz 8960 chips e três vezes mais memória HBM.

	v4	v5e	v5p
Chips por unidade	4.096	256	8.960
Largura de Banda Interconectores	2.400 Gbps	1.600 Gbps	4.800 Gbps
BF16 TFLOPS	275	197	459
Memória HBM	32 GB	16 GB	95 GB
Largura de Banda HBM	1.228 GB/s	820 GB/s	2.795 GB/s

Figura 6.1: Comparativo entre TPUs Google para cargas de trabalho em IA e LLM

Mas no fim, o que significam esses valores? A Google promete que a nova TPU consegue escalar o tempo de treinamento, sendo até $4X$ mais rápida que TPUs mais baratas, como a $V4$, devido ao dobramento no tamanho de operações em ponto flutuante que são entregues.

Capítulo 7

Consumo de Energia

Atualmente, TPUs são encontradas com maior frequência em data centers, e são um produto extremamente utilizado não só pela Google, como também por empresas e usuários que dependem de seus serviços de nuvem.

Para avaliar, primeiramente, a performance/watt da TPU, e comparar com outras arquiteturas como GPU, se utiliza as métricas "total"(que calcula a energia consumida pela TPU/GPU juntamente a CPU, quando calcula performance/watt) e "incremental"(que subtrai a energia consumida pela CPU). A figura abaixo ilustra bem os resultados obtidos pela Google, que mostram uma performance de 17 a 34 vezes melhor que a GPU no modelo total, e de 25 a 29 vezes melhor que a GPU no modelo incremental.

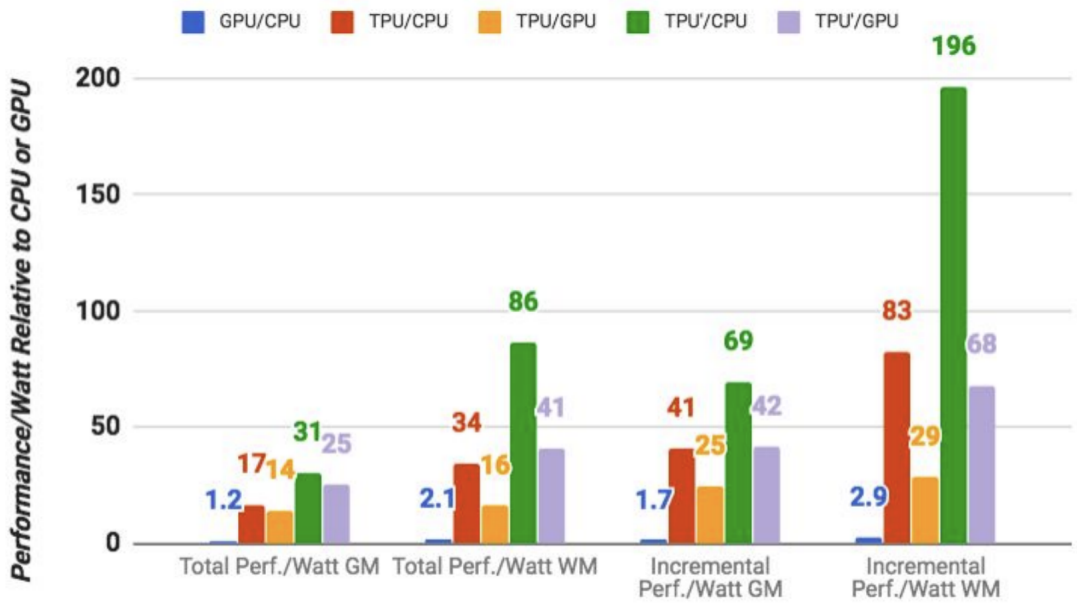

Figura 7.1: Performance/Watt para operações de médias geométricas e ponderadas (GM e WM, respectivamente)

Tendo esses consumos em vista, uma das medidas adequadas para se medir juntamente ao surgimento de uma nova arquitetura é o TDP (Thermal Design Power), que afeta diretamente o custo de energia gasto, tendo em vista que é necessário que a unidade receba energia e resfriamento suficientes. Tendo em vista que servidores não estão em funcionamento durante 100% do tempo, é interessante que a energia consumida por essas máquinas seja proporcional ao seu tempo de uso.

A fim de medir tal consumo e avaliar a validade energética do uso de TPUs em detrimento de GPUs, a figura a seguir compara a energia gasta pelo servidor dividido pelo número de dias, variando o workload para processar CNN0, utilizando o mesmo batch para todos os testes.

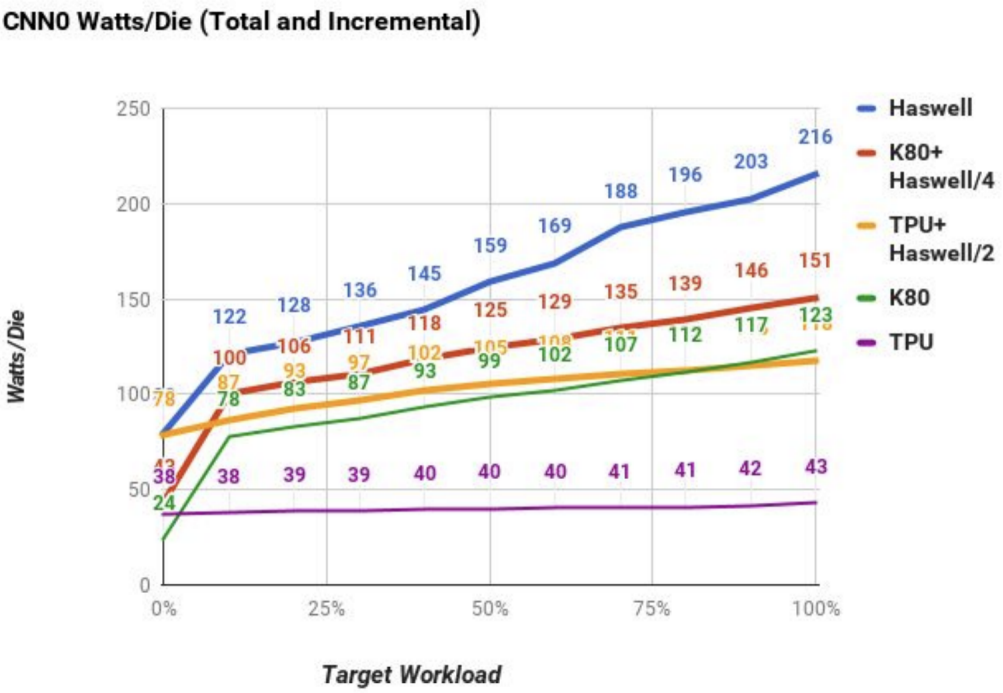

Figura 7.2: Performance/Watt para operações de médias geométricas e ponderadas (GM e WM, respectivamente)

Pelos resultados obtidos, vemos que as TPUs conseguem consumir, no geral, menos energia que as GPUs, porém há um trade-off grande, pois a quantidade de energia consumida, mesmo com pouca operação na máquina, ainda é muito similar ao consumo com a máquina funcionando em 100%. Ou seja, quando a TPU e GPU estão totalmente carregadas, o servidor de CPU gasta 52% da energia da GPU e 69% da energia da TPU. Nesse sentido, a TPU ainda gasta mais energia proporcionalmente, pois realiza muito mais tarefas que a GPU, porém seus gastos em valor absoluto ainda são inferiores, e justificam o uso desse hardware.

Capítulo 8

Considerações Finais

Bibliografia

TPU e Arquitetura
Comparativos oficiais em modelos de IA (TPU vs GPU)
TPU v5p
Conceitos básicos
Wikipedia, tem tudo
Paper original da TPU