Escuela Politécnica Nacional

Nombre: Francisco Ulloa
Fecha: Quito, 28 de enero de 2026
Tema: Factoreo en transformers
Repositorio:
https://github.com/Fu5CHAR/Metodos_numericos_2025B_Ulloa-Francisco/tree/main

Comparación de arquitecturas NVIDIA: Hopper vs Blackwell

1. Diferencias generales entre Hopper y Blackwell

Arquitectura Hopper

- Lanzada en 2022–2023.
- Orientada a **IA generativa y computación de alto rendimiento (HPC)**.
- Introduce el **Transformer Engine**, que usa múltiples precisiones (FP16, BF16, FP8).
- Incorpora **Tensor Cores de cuarta generación**.
- Soporta aceleración avanzada para entrenamiento de modelos grandes (LLM).

Arquitectura Blackwell

- Lanzada en 2024–2025.
- Sucesora directa de Hopper.
- Incorpora **Tensor Cores de quinta generación**.
- Introduce soporte nativo para **precisiones ultra-bajas (FP6 y FP4)**.
- Mayor eficiencia energética, más ancho de banda y mejor escalabilidad.
- Pensada para **IA generativa masiva e inferencia a gran escala**.

Comparación resumida

Característica	Hopper	Blackwell
Generación Tensor Core	4ª	5ª
Enfoque principal	Entrenamiento IA + HPC	IA generativa eficiente
Precisión mínima	FP8	FP4
Eficiencia energética	Alta	Muy alta
Escalabilidad	Alta	Superior

---

2. Diferencia entre FP32 y TF32 (a veces llamado TP32)

FP32 (Floating Point 32)

- Formato estándar IEEE 754.
- Usa **32 bits**:
  - 1 bit de signo
  - 8 bits de exponente
  - 23 bits de mantisa
- Alta precisión numérica.
- Usado tradicionalmente en computación científica.

TF32 (TensorFloat-32)

- Formato introducido por NVIDIA para **Tensor Cores**.
- Mantiene:
  - 8 bits de exponente (mismo rango que FP32)
- Reduce:
  - Mantisa a ~10 bits
- No es un formato de almacenamiento, sino **un modo de cálculo acelerado**.
- Aumenta significativamente el rendimiento con mínima pérdida de precisión.

Comparación FP32 vs TF32

Formato	Bits totales	Exponente	Mantisa	Precisión	Rendimiento
FP32	32	8	23	Alta	Medio
TF32	32	8	~10	Media	Muy alto

---

3. Representaciones de datos soportadas

Hopper

- **CUDA Cores**
  - FP64
  - FP32
  - FP16
  - BF16
  - INT8
- **Tensor Cores**
  - FP64
  - TF32
  - FP16

- BF16
- FP8
- INT8

Blackwell

- **CUDA Cores**
  - FP64
  - FP32
  - FP16
  - BF16
- **Tensor Cores**
  - FP64
  - TF32
  - FP16
  - BF16
  - FP8
  - FP6
  - FP4
  - INT8

Comparación de soporte

Precisión	Hopper	Blackwell
FP64	✔	✔
FP32	✔	✔
TF32	✔	✔
FP16	✔	✔
BF16	✔	✔
FP8	✔	✔
FP6	✖	✔
FP4	✖	✔
INT8	✔	✔

---

4. ¿Por qué la nueva arquitectura prefiere menor precisión?

1. Mayor rendimiento computacional

- Menos bits permiten **más operaciones por ciclo de reloj**.

- Se incrementan los FLOPS efectivos.

2. Menor uso de memoria y ancho de banda

- Datos más pequeños reducen:
  - Acceso a memoria
  - Consumo de ancho de banda
- Factor crítico en modelos grandes.

3. Tolerancia de los modelos de IA

- Redes neuronales profundas **no requieren precisión completa** en la mayoría de capas.
- La reducción de precisión no degrada significativamente la calidad del modelo.

4. Mejor eficiencia energética

- Menor consumo por operación.
- Reducción de costos operativos en centros de datos.

Conclusión

- **Hopper** introduce la aceleración avanzada con FP8 y TF32 para IA moderna.
- **Blackwell** extiende este enfoque hacia **precisiones ultra-bajas (FP4, FP6)**.
- El uso de menor precisión permite:
  - Más rendimiento
  - Menor consumo energético
  - Escalabilidad para IA generativa masiva