Apuntes Martes 27/05/2025

Luis Carlos N. Todd c.2022212158

***Abstract*—This document presents a detailed overview of quantization techniques in deep learning. It covers the mathematical foundations and differences between symmetrical and asymmetrical quantization, including formulas and practical considerations. Techniques such as dynamic quantization, calibration, post-training quantization (PTQ), and quantization-aware training (QAT) are explained alongside the role of observers. Additionally, the document introduces unsupervised learning, emphasizing clustering methods and dimensionality reduction techniques like t-SNE. It explores clustering strategies such as K-means and hierarchical methods.**

***Index Terms*—Quantization, Dequantization, Dynamic Quantization, Calibration, PTQ, QAT, Unsupervised Learning, Clustering.**

I. Quantization

Como se explicó la clase pasada la técnica de quantization se utiliza para reducir el tamaño en memoria de los modelos de deep learning usando otra representación de los pesos. Modelos como LLaMA 2 tiene 70 mil millones de parámetros. También, hacer operaciones de punto flotante es lento.

$$\frac{7 * 10^9 * 32}{8 * 10^8} = 28GB \tag{1}$$

(1) Tamaño en GigaBytes de LLaMA 2 con pesos de 32 bits.

Se reducen el número de bits requeridos para representar cada parámetro. Se pasa de punto flotante a entero Forma de compresión de un modelo a 8, 5, 2 y 1 bit.

A. Aclaratoria

Quantization no significa aplicar técnicas de redondeo a nuestros pesos y convertirlos a enteros. La técnica tiene otros cálculos.

B. Ventajas

Menor Consumo de Memoria: Al momento de cargar los modelos. Se pueden usar devices.

Menor tiempo de inferencia: Datos simples

Menor consumo de energía: Costo de computación más barato.

C. Tamaños y A2

Generalmente:

Un long usa 8 bytes.

Un int usa 4 bytes.

Un short usa 2 bytes.

La computadora utiliza complemento A2 para representar negativos.

D. Representación Punto Flotante

Ver estándar IEEE-754.

E. Redes neuronales - Pesos

Matrices de pesos son representadas con punto flotante.

Queremos no perder calidad después de haber entrenado el modelo.

Qué queremos cuantizar? Si tengo wx+b, se tienen que cuantizar las 3 variables *w, x y b*. La salida se tiene que decuantizar para que sea procesada por la siguiente capa.

Al cuantizar pierdo información o precisión.

F. Asimétrico

Se calculan β y α que son los valores menores y mayores de un tensor respectivamente. Se tiene un límite de representación entre 0 a 255. El valor de cero es representado por z en la nueva representación.

Fórmula Quantization

$$x_q = clamp(|\frac{x_f}{s}| + z; 0; 2^n - 1) \tag{2}$$

Fórmula Decuantization

$$x_{fd} = s(x_q - z) \tag{3}$$

Estas son las fórmulas de quantization asimétrico, donde x_f es el valor flotante por cuantizar, x_q es el valor cuantizado, x_{fd} es el valor decuantizado (con pérdida de precisión), n es el número de bits utilizado para la cuantización y s es el parámetro de escalado.

Fórmula de S: Parámetro de Escalado

$$s = \frac{\alpha - \beta}{2^n - 1} \tag{4}$$

G. Simétrico

Valor mayor absoluto de todo el tensor. Se tiene límite de -127 a 127. El valor de cero se mantiene en ambas representaciones.

Fórmula Quantization

$$x_q = clamp(|\frac{x_f}{s}|; -(2^{n-1} - 1); 2^{n-1} - 1) \tag{5}$$

Fórmula Decuantization

$$x_{fd} = sx_q \tag{6}$$

Fórmula de S: Parámetro de Escalado

$$s = \frac{|\alpha|}{2^{n-1} - 1} \tag{7}$$

H. Dynamic Quantization

Queremos ejecutar todas las operaciones de enteros. Cómo cuantizamos la entrada *X*? Las activaciones pueden ser cuantizadas en el momento, calculando el alfa y el beta. "Quantization on the flight". Una vez cuantizado el resultado Y_f será de valores enteros. Para la siguiente capa se requiere los valores decuantizados (flotantes). Cómo se calculan sin su escalador y sin su cero?

I. Calibration

Hacemos inferencia de las salidas usando algunas entradas y observando cuáles son las salidas típicas. Obtenemos un alfa y beta razonables para calcular el escalador s y el valor de z.Se realiza en la etapa post-training quantization.

J. Estrategia de selección del rango

Usar Min-Max no es la única técnica de selección de rangos. Min-Max es sensible a outliers. Esto empeora el proceso de quantization y agrega mucho error por el proceso. Para eso se puede usar el percentil.

$$alfa = percentil(V, 99\%) \tag{8}$$

$$beta = percentil(V, 1\%) \tag{9}$$

K. Quantization Granularity (Convolución)

Las convoluciones están hechas de muchos filtros. Aprendidos con valores y distribuciones distintas.

L. Post Training Quantization (PTQ)

Utilizan datos nunca vistos por el modelo.

Observers: Encargados de obtener información estadística de cada capa. Calculan los parámetros s y z. Permite cuantizar el modelo.

Se le agregan observers al modelo y se calibran con datos no antes vistos o datos de producción. Después de calibrar se puede quantizar el modelo.

M. Quantization Aware Tunning (QAT)

Se hace el proceso de quantize - dequantize al momento de entrenar. Por lo tanto, el valor de pérdida se va a encargar de calibrar el modelo haciéndolo más robusto. Los observers se agregan antes de entrenar el modelo.

N. Notas del Notebook: QPT

Se tiene el modelo de simplenet y se entrena normalmente. Después instancio un modelo de simplenet con observers en el modelo, cargo los pesos del modelo pasado, hago el convert, se pone en modo .eval() y se usa con el testing set o datos de producción. Así se calibra. El accuracy del modelo se mantiene en 0.97.

O. Notas del Notebook: QAT

Se ponen los observers y en el forward se pone quantize al principio y dequantize al final. Se carga la configuración y se entrena. Ahora sí se puede hacer el convert después de entrenar y usarlo. El modelo ya está calibrado, entonces se hace el convert y se puede utilizar.

II. Unsupervised Learning

A diferencia del supervised learning donde se tenían los datos de input x y su respectivo label y, en los algoritmos de unsupervised learning solo se tiene un dataset con inputs sin etiquetas.

Se tiene $(Xi, {}_{-i}^{N})$

Sirve para reducción de dimensionalidad por ejemplo con los autoencoders. Detección de outliers.

Se usan estrategias de clustering para visualizar los datos. Con los clusters se pueden agrupar los datos para intentar etiquetarlos.

Se pueden ver los algoritmos en la página de scikit-learn. Hay muchos sabores de los algoritmos de clustering. Verlo en 2 dimensiones es más fácil pero

A. Visualización mediante t-SNE

Es un paper de una técnica de visualización y clustering de datos.

B. Clustering

Procedimiento para identificar grupos en los datos, basado en la similitud de los features. Se divide el espacio del conjunto de datos en N grupos, lo más homogéneo posible. No hay forma perfecta de clustering, solamente diferentes técnicas.

Hay múltiples definiciones de clustering:

* Proceso de encontrar grupos en los datos.
* Dividir los datos en grupos homogéneos.
* Dividir los datos en grupos, donde los features de cada uno de los grupos están lo más cerca posible (similares).
* Dividir los datos en grupos, donde los features de cada uno de los grupos están lo más cerca posible (similares) y los features de diferentes grupos están lo más lejos posible.

Las definiciones son muy amplias y pueden variar dfependiendo del dominio del problema. Hay muchos artículos al respecto. Clusters en alta dimensionalidad pueden no tener sentidos para los seres humanos.

C. Principios y Técnicas

Se basa en similaridad o distancia entre los objetos del dataset. Similaridad típicamente se saca con la distancia entre los vectores de los features. No usar en datos categóricos.

Hay diferentes tipos de distancias (Euclidiana, Manhattan, Similitud de Cosenos, Hamming, Minkowski).

La distancia de coseno o similitud de coseno es lo que se utiliza en las DBs vectoriales.

Centralidad de clúster Basados en puntos centrales (centroides). Representa el promedio o la ubicación central de todos los puntos en un clúster. Algunos algoritmos como K-Means calcula iterativamente. Otros usan los puntos centrales para unir o dividir de los clusters (Jerarquicos).

Minimizar una función objetivo La función objetivo dtermina la mejor disposición del clúster. K-means busca el punto medio entre todos.

K-means en general funciona bien pero para datos con alta dimensionalidad sirve más Birch u otros

D. Tipos de Clustering

Jerárquicos: Se van haciendo divisiones jerárquicas entre los datos.

Deterministico: La muestra pertenece a un cluster o no pertenece.

Probabilístico: La muestra tiene probabilidades de pertenecer a los clusters.

E. K-Means

Se tienen N centroides colocados en posiciones randoms. Calculos las distancias de cada punto a los centroides y etiqueto según el más cercano. Después recalculo la posición de los centroides según el valor promedio de todos los puntos con esa etiqueta y después vuelvo a calcular la distancia entre todos los puntos, etc...