

## 基於 DeepSpeed 架構之訓練及資源優化

專題編號：112-CSIE-S015

執行期限：111 年第 1 學期至 112 年第 1 學期

指導教授：劉傳銘

專題參與人員：109590014 沈煒翔

109590033 吳聖翊

### 一、摘要

深度學習在當今的科學和工程領域中扮演著關鍵的角色，然而，訓練大型深度學習模型的需求對於速度和硬體資源的要求日益增加，這使得對於普通人而言變得難以企及。在這個背景下，我們專注於分析和探討 DeepSpeed 這一工具的潛力。

DeepSpeed 被譽為一個能夠在資源有限的環境下訓練大型深度學習模型的解決方案。然而，我們發現原始碼直接優化的資源有限，這促使我們啟動了這個專題計畫。我們的目標是透過深入分析 DeepSpeed 相關原始碼，尋找優化的機會，以提高訓練速度和降低硬體資源的需求。這將有助於普通研究人員和開發者更容易訓練大型深度學習模型，並使其變得更加可及。

**關鍵詞：**DeepSpeed、深度學習、硬體資源

### 二、緣由與目的

深度學習的發展速度驚人，特別是在模型大小的增長方面，這增長幅度遠超過硬體資源的提升。在不斷推陳出新的深度學習模型中，模型參數的數量已經變得巨大，這對硬體資源的消耗提出了嚴峻挑戰。為了應對這個問題，我們特別關注了 Microsoft 提出的 DeepSpeed 訓練框架，其中包含了 ZeRo 技術[1]，這個技術在減少 VRAM 使用方面取得了顯著的成效。然而，它同時引入了 DRAM 和 VRAM 之間更多的溝通，甚至 ZeRo-Infinity 更進一步利用 NVMe 裝置進行資源的優化[2][3][4]。

然而，實際上 DeepSpeed 的訓練框架並未像相關論文中所描述的那樣進行了如

此多的優化。

因此，我們的研究目的在於設計一種方法，通過深入分析 DeepSpeed 訓練框架的原始碼，檢測現有框架中的潛在優化機會，並尋找方法進行實質性的優化。我們的最終目標是降低硬體資源在深度學習訓練過程中的消耗，同時保持高效率。透過這項研究工作，我們期望為深度學習社區提供更有效且可持續的訓練框架，使更多人能夠輕鬆參與這一領域的研究和應用。

DP	7.5B Model (GB)			128B Model (GB)			1T Model (GB)		
	$P_{os}$	$P_{os+g}$	$P_{os+g+p}$	$P_{os}$	$P_{os+g}$	$P_{os+g+p}$	$P_{os}$	$P_{os+g}$	$P_{os+g+p}$
1	120	120	120	2048	2048	2048	16000	16000	16000
4	52.5	41.3	<b>30</b>	896	704	512	7000	5500	4000
16	35.6	<b>21.6</b>	7.5	608	368	128	4750	2875	1000
64	<b>31.4</b>	16.6	1.88	536	284	<b>32</b>	4187	2218	250
256	30.4	15.4	0.47	518	263	8	4046	2054	62.5
1024	30.1	15.1	0.12	513	257	2	4011	2013	<b>15.6</b>

圖一、ZeRo 官方優化數據圖

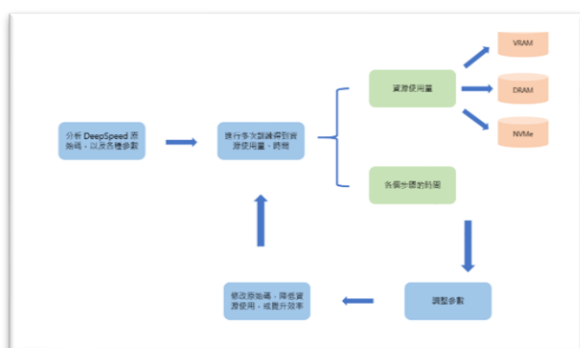
### 三、研究報告內容

#### (一)、研究方法及步驟

本研究首先專注於深入理解 DeepSpeed 框架的核心結構和運作方式。包括研究其支援的模型類型、訓練方法、以及參數設定的選項。探索 DeepSpeed 框架中降低硬體資源使用率的方法，特別關注 VRAM、DRAM 和 NVMe 的使用。包括模型平行化、梯度累積、混合精度訓練等技術。理解這些方法如何應用於 DeepSpeed 上。

選擇 LLAMA 7B 和 BLOOM 560M 這兩個具有不同特性的模型，作為測試用例。根據這兩個模型的需求，設計測試方案來評估 DeepSpeed 框架的性能和資源使用。使用 DeepSpeed 框架，以默認的參數設定進行初

步測試，以了解原始性能。記錄模型訓練過程中的硬體資源使用情況，包括 VRAM、DRAM 和 NVMe 的使用。在多次測試中，調整 DeepSpeed 框架的參數設定，以尋找性能和資源使用的最佳平衡。這可能涉及到學習速率、批次大小、模型架構等參數的調整。根據測試結果，進一步修改 DeepSpeed 框架的原始碼，以提高速度並降低 VRAM、DRAM 和 NVMe 的使用。這可能包括針對模型訓練過程中的資源管理進行改進。分析優化後的結果，並將其與原始框架的性能進行比較。評估改進的效果，確保已經實現了硬體資源的節省和訓練效率的提升。



圖二、研究步驟圖

## (二)、研究結果

經過多次實驗，我們發現實際使用 DeepSpeed 框架時，其性能並未達到論文中所描述的優異效果。在我們的測試環境中，一開始雖然模型可以運行，但仍然面臨資源不足的問題，無法輕易調整 Batch size 和 sequence length。因此，我們集中於初始化記憶體的部分進行優化。具體而言，DeepSpeed 在初始化時會使用最大的緩衝區大小，這導致了記憶體的浪費。為了解決這個問題，我們對 DeepSpeed 的初始化過程進行了分析，並將緩衝區大小分成兩組，一組是大的，一組是小的。這樣的優化使記憶體使用量減少了約 20%，同時確保模型可以正常運行。

此外，我們的研究中使用了 NVMe 進行訓練，並專注於提高讀寫速度。為了實現這一目標，我們考慮將操作系統中的檔案

系統進行調整，以建立一個更精簡的版本。這種調整預期將有助於提升 NVMe 的效能，進一步改進模型訓練的速度。總之，我們的研究結果表明，在實際應用 DeepSpeed 框架時，需要進一步的優化，特別是在資源管理方面。這些改進對於確保模型的穩定運行以及減少資源浪費都具有重要意義，並有望提高深度學習模型的效率。

## 四、參考文獻

- [1] Olatunji Ruwase, Samyam Rajbhandari Jeff Rasley, and Yuxiong He. *ZeRO: Memory Optimizations Toward Training Trillion Parameter Models*. Oct. 2019.
- [2] Jeff Rasley, Samyam Rajbhandari Olatunji Ruwase, and Shaden Smith Yuxiong He. *ZeRO-Infinity: Breaking the GPU Memory Wall for Extreme Scale Deep Learning*. Nov. 2021.
- [3] Guanhua Wang, Heyang Qin, Sam Ade Jacobs, Connor Holmes, Samyam Rajbhandari, Olatunji Ruwase, Feng Yan, Lei Yang, Yuxiong He. *ZeRO++: Extremely Efficient Collective Communication for Giant Model Training*. June 2023.
- [4] Jie Ren, Samyam Rajbhandari, Reza Yazdani Aminabadi, Olatunji Ruwase, Shuangyan Yang, Minjia Zhang, Dong Li, Yuxiong He. *ZeRO-Offload: Democratizing Billion-Scale Model Training*. Jan. 2021.