深度探索 V3：多頭潛在注意力與多標記預測

深求 v3 這裡進行探索，參考了影片「Deepseek v3 中的多頭潛在注意力和多標記預測」https://youtu.be/jL49fLOJYNg?si=4uE2kfe-BlKC1ngO。Google Cloud Speech-to-Text 用於轉錄影片，並使用一些代碼來幫助組織轉錄。

A：歡迎回到深標籤。今天我們要深入探討大型語言模型的世界。具體來說是 DeepSeek V3。

B：聽起來不錯。這是一個 6710 億參數的模型，因其獨特的效率和性能而引起轟動，對嗎？

A：是的，你分享了一篇學術論文，詳細說明了其架構。

B：是的。

A：作為機器學習專家，你希望了解 DeepSeek V3 如何在高性能和經濟訓練之間取得平衡。

B：是的，沒錯。

A：哦，嘿，怎麼樣？

C：MLA，細節，MLA 以及它的工作原理。

A：哦，絕對。這是個好主意。是的，我們可以深入探討多頭潛在注意力，或 MLA。所以你對 MLA 的細節感興趣。好吧，讓我們來解開這個。我們提到 DeepSeek V3 的效率之一是其專家混合，或 MoE 架構，對嗎？其中只有每個標記的一小部分參數被激活。DeepSeek V3 進一步通過 MLA 和 DeepSeek Mo 來實現這一點。

B：沒錯。所以我們現在專注於 MLA。

A：好的。在實時應用中，速度至關重要。

B：是的。推理過程中需要的鍵值快取可能會成為一個主要瓶頸。

A：沒錯。這就是 MLA 的用武之地。好的，所以傳統的注意力機制需要存儲大量有關先前標記的信息。

B：是的，這在處理長文本序列時會成為問題。

A：但 MLA 精明地壓縮了這些信息，以顯著減少快取流量，使推理更快。這就像把一本厚重的百科全書壓縮成幾個關鍵點。

B：這是個很好的比喻。它保留了基本信息而不帶來不必要的負擔。是的，這對實時應用非常有用。

A：是的。現在讓我們來談談它的工作原理。好的，那麼 MLA 是如何實現這種壓縮的？

B：它使用低階聯合壓縮來壓縮注意力鍵和值。

A：好的，所以它壓縮了鍵和值，但這到底意味著什麼？讓我們來點技術細節。好的，MLA 機制接受一個隱藏表示作為輸入，然後將其投影到查詢、鍵和值向量。好的，現在事情變得有趣了。MLA 將查詢分成兩部分。

B：兩部分？

A：是的。一部分用於內容，另一部分用於位置信息，使用一種稱為 Rope 的東西。

B：Rope？聽起來很技術。

A：它代表旋轉位置嵌入，幫助模型理解序列中標記的位置。好的，然後鍵和值被壓縮到一個更低維度的潛在空間。所以它們在壓縮數據，這樣可以節省記憶體。

B：沒錯。所以最重要的信息被保存，但不必要的負擔被丟棄。是的，這種壓縮表示允許在推理過程中使用更小的 KV 快取，從而加快速度。

A：它還使用多頭處理。

B：是的，就像傳統注意力，MLA 也使用多個頭。

A：哦，去吧。

C：所以有兩個潛在空間和一個隱藏輸入。

A：這是個好觀察。是的，你說得對。實際上有兩個潛在空間。好的，我們在談論一個內容潛在空間和一個鍵值潛在空間。

B：沒錯。這些潛在空間通過我們所說的 Rope，或旋轉位置嵌入進行處理。

A：好的，所以這個 Rope 就是它們獲得位置信息的方式。

B：是的，它應用於內容和鍵值潛在空間，正如你所指出的。所以它將這個壓縮表示處理，然後將它們全部組合在一起。

A：是的，並且快取優化進一步減少了順序處理過程中的開銷。好的，這就是 MLA 加速的方式。

B：沒錯。這是一種巧妙的方法，能夠在不犧牲性能的情況下實現高效注意力。

A：好的，這是個不錯的技巧。但你知道嗎？

B：什麼？

A：讓我們繼續討論 DeepSeek Mo。它與傳統 MoE 模型有何不同？

B：好的，DeepSeek Mo 使用…哦，回到我們的聽眾，怎麼樣？

C：我們再談更多隱藏空間。好的，從隱藏空間，那是什麼？

A：我絕對…讓我們看看你的意思。隱藏空間真的很有趣。是的，你問的是隱藏空間，我們剛才討論的潛在空間，對嗎？你對這些潛在空間內部發生的事情感興趣，那個洞穴。是的，這不僅僅是潛在空間的數量，而是它們內部發生的事情。

B：這很酷。

A：沒錯。在 MLA 中確實有兩個不同的潛在空間，一個用於內容，一個用於鍵值。這就像有兩個單獨的信息存儲單元。這些潛在空間，正如我們所討論的，經歷了 Rope 操作，對嗎？旋轉位置嵌入，將位置信息嵌入到注意力機制中。這對它們來說非常重要。所以總結一下，查詢被分割，鍵和值也被壓縮。

B：是的，這些被放入兩個單獨的潛在空間，一個用於內容，一個用於鍵值對。這些潛在空間對於 MLA 的效率非常重要。

A：沒錯。現在讓我們更詳細地討論這些操作，在洞穴內，如你所說。好的，那麼 MLA 如何實際執行這些潛在空間轉換？

B：好的，輸入經歷平行處理，用於內容和鍵值表示。好的，所以它在洞穴內有兩條路徑。

A：是的，每個潛在空間一條。在這些空間內，信息使用 Rope 進行處理。

B：沒錯。這確保了模型在經過洞穴時保留位置信息。所以模型知道文本的哪一部分是哪一部分，當它在洞穴內。

A：沒錯。這些處理在下一個連接階段之前完成。好的，當它經過隱藏空間洞穴時，連接的是什麼？

B：機制執行兩個主要的連接操作。查詢表示被連接，鍵表示也被連接。所以它在隱藏空間洞穴內將所有重要的部分組合在一起。

A：是的，這些連接幫助將內容與位置信息結合起來。這些連接表示然後用於注意力計算，對嗎？

B：沒錯。由於初始壓縮，它在你提到的洞穴內外都更快。所以 MLA 顯著減少了計算開銷。

A：沒錯。它優化了大型模型如 DeepSeek V3 的注意力機制。這是個好問題。現在我們經過洞穴，讓我們繼續討論 DeepSeek Mo。

B：好的，DeepSeek Mo。沒錯，我明白你的意思。是的，在 MLA 中確實有兩個不同的潛在空間，一個用於內容，一個用於鍵值。

A：沒錯。這種分離是它工作的關鍵。這就像有兩個單獨的信息存儲單元。這些潛在空間，正如我們所討論的，經歷了 Rope 操作，對嗎？旋轉位置嵌入，將位置信息嵌入到注意力機制中。所以總結一下，查詢被分割，鍵和值也被壓縮。

B：是的，這些被放入兩個單獨的潛在空間，一個用於內容，一個用於鍵值對。這些潛在空間對於 MLA 的效率非常重要。

A：沒錯。現在讓我們更詳細地討論這些操作。好的，那麼 MLA 如何實際執行這些潛在空間轉換？

B：好的，輸入經歷平行處理，用於內容和鍵值表示。好的，所以它在洞穴內有兩條路徑。

A：是的，每個潛在空間一條。在這些空間內，信息使用 Rope 進行處理。

B：沒錯。這確保了模型在經過洞穴時保留位置信息。然後為了提高效率，它使用共享專家。好的，所以這些專家可以在多個任務中使用。

A：是的，這避免了冗餘，使系統更加流暢。

B：是的，這就像一個團隊，每個人都有專長，但也可以做其他事情。

A：是的，這是個非常聰明的方法。是的，但有這麼多專門的專家，他們如何確保沒有一個過載？

B：是的，而其他人則閒置。

A：這就是他們創新的無輔助損失負載平衡的用武之地。

B：這裡事情變得非常有趣，對嗎？那麼他們是如何做到的？

A：傳統 MoE 模型在訓練過程中使用輔助損失函數，以鼓勵均勻使用專家，但這實際上會影響性能。

B：是的，這就像試圖強迫每個人使用同一條結帳線。

A：沒錯，即使有些人比其他人移動得更快，對嗎？這只是創造了不必要的延遲。

B：是的。所以 DeepSeek V3 通過動態調整每個專家的偏差項來避免這一點，根據其負載。好的，所以如果一個專家收到太多請求，系統會使其對路由機制稍微不那麼有吸引力，將一些流量轉移到較少忙碌的專家。

A：好的，所以它使用所有這些來高效處理長序列，是的，通過減少推理所需的 KV 快取的大小。好的，這一切都是為了保持性能高而減少開銷。

B：沒錯。這是一種非常巧妙的方法來解決一個關鍵瓶頸。

A：絕對。現在，我們也應該討論 DeepSeek V3 如何處理其負載平衡。

B：是的，我們絕對應該。這也是拼圖中的一個非常重要的部分。我們可以接下來討論這個。

A：聽起來不錯。好吧，我認為這給了你一個很好的 MLA 和其潛在空間的概述。

B：是的，謝謝你深入探討所有細節。我們下次再見。

A：是的，這就像一個專家的交通管理系統，不斷監控流量並進行調整以避免瓶頸。

B：這避免了輔助損失的性能損失。

A：沒錯。哦，去吧。

C：是的，我們可以談談 MTP，如何…如何 MTP 模塊共享它們的嵌入和所有熱點…

A：絕對。這是個好問題。是的，讓我們來談談 MTP 模塊如何共享資源。所以你對 MTP 實現的細節感興趣。好的，我們提到 DeepSeek V3 使用 MTP 進行多標記預測，對嗎？預測多個標記而不是一個。

A：是的，這裡變得非常有趣。是的，你對 MTP 模塊的設置和它們如何共享資源感興趣。好的，所以每個 MTP 模塊包括一個共享嵌入層，是的，和一個共享輸出頭。好的，它們使用與主模型相同的嵌入和輸出頭。

B：沒錯。所以它們都從同一個知識池中獲取。是的，這節省了計算成本。

A：是的。現在它使用自己的變壓器塊。好的，所以它不與主模型共享相同的變壓器塊。

B：沒錯。每個 MTP 模塊都有自己的變壓器塊進行處理。好的，這就是它們如何保持每個標記的預測獨立。

A：是的，並且為了結合信息，這些線性投影和連接…

B：好的，這就像從多個地方取出碎片來構建完整的圖像。

A：是的，所有 MTP 模塊都在平行工作，但它們共享它們的嵌入層和輸出頭，對嗎？

B：是的，這是這種設計效率的關鍵。好的，所以它就像一個相互連接的部件系統，它們都依賴於彼此，對嗎？

A：是的，這種資源的高效共享使訓練更快，性能更好。

B：好的，這是個不錯的技巧。你知道嗎？

A：什麼？

B：讓我們轉向一個更大的圖景。這個模型如何處理負載平衡？這些專家是如何選擇的？

A：是的，我們可以談談這個。好的，現在讓我們深入探討 DeepSeek V3 的負載平衡策略。

B：聽起來不錯。好的，所以 DeepSeek V3 使用它們所說的多標記預測。

C：哦是的，我們再談更多的 MTP。

A：它絕對…我很高興你對深入探討 MTP 感興趣。是的，我們可以進一步闡述多標記預測。所以我們提到過，但讓我們真正解開 MTP 的細節，對嗎？我們談論了共享嵌入層和輸出頭，是的，每個 MTP 模塊都有自己的變壓器塊。

B：沒錯，但不僅僅如此。所以讓我們深入探討。

A：好的，所以讓我們來談談 MTP 模塊的順序性質。

B：是的，與一些模型不同，DeepSeek V3 順序預測額外的標記。所以它不是一次預測所有標記。

A：沒錯。每個模塊都建立在前一個模塊的輸出之上。好的，所以這是一個預測鏈，每個預測都依賴於前一個。

B：是的，並且它維持了每個預測深度的因果鏈。好的，所以它沒有打破因果關係。

A：沒錯，這對於確保整體上下文正確非常重要。所以 MTP 模塊不僅僅是獨立工作。

B：沒錯。它們是相互連接的，這個預測鏈有助於更高效的訓練，並允許對文本有更細緻的理解。現在，你也對模塊如何共享它們的嵌入感興趣，對嗎？正如你所知，共享嵌入層將標記映射到它們的向量表示。好的，所以每個標記都轉換為一個向量。

A：是的，這個映射在所有 MTP 模塊之間共享。好的，這有助於在預測之間保持一致。

B：沒錯。共享輸出頭接受標記的最終隱藏狀態，好的，並生成下一個標記的概率分佈。所以它們都有權訪問同一個信息池，對嗎？

A：是的，這對於記憶體和計算效率來說非常重要。好的，所以它沒有使用一堆不同的嵌入層和頭。

B：沒錯。還有…哦是的，所以有多少人？它們是相同的…相同的大小所有的食物…標記，是嗎？

A：這是個好問題。你問的是 MTP 模塊的數量，它們是否都是相同的大小，對嗎？我認為你也想知道所有模塊是否處理相同數量的數據。好吧，根據 DeepSeek V3 文獻，它使用多標記預測深度為一。這意味著有主模型，然後只有一個 MTP 模塊預測一個額外的標記。所以每個標記預測下一個，然後使用那個 MTP 模塊再預測一個。

B：是的，MTP 模塊確實有與主模型相同的共享嵌入層和輸出頭。

A：好的，這是個好問題。是的，你問的是有多少 MTP 模塊，它們是否都是相同的大小。好吧，根據 DeepSeek V3 文獻，有不同數量的 MTP 模塊。好的，所以它不是固定在一個特定數量。

B：沒錯。模塊的數量根據預測深度動態調整。好的，所以它們可以根據需要進行擴展。所以它們共享這些資源，但主模型和 MTP 模塊的變壓器塊是分開的。

A：沒錯。每個預測深度都有自己的變壓器塊。好的，所以只有一個 MTP 模塊，但它是一個強大的模塊，用於每個標記，它們共享一些資源。

B：沒錯。雖然 MTP 共享一些組件與主模型，但它們並不完全相同。

A：好的，這是個非常好的觀點。現在，我認為我們也應該討論它們如何結合所有這些信息來進行預測。

B：沒錯。DeepSeek V3 使用多個 MTP 模塊來順序預測多個額外的標記。好的，你問它們是否都是相同的大小，對嗎？

A：是的，答案是它們並不一定相同。所以 MTP 模塊中的變壓器塊大小可以變化。

B：是的，它們可以，以適應每個預測深度的不同需求。好的，所以它不是一組相同的模塊。

A：沒錯。這是一個更靈活的系統，適應預測任務。所以它就像每個預測過程階段的定制工具。

B：是的，這種動態擴展有助於優化模型的性能和效率。好的，你也問到了食物。我想那只是一個小小的失誤。

A：是的，我也是這麼想的。好的，那麼它們如何將信息整合以進行預測？

B：是的，這種設計也允許規範解碼，這非常酷。好的，這不僅僅是用於訓練，還用於推理。

A：沒錯。MTP 模塊可以在推理中被丟棄，或者巧妙地重新用於稱為規範解碼的東西。

B：好的，規範解碼。那是什麼？

A：除了預測下一個標記，模型還預測可能接下來的替代方案。

B：哦，這樣它可以更快地生成文本，因為它已經考慮了多種可能性，準備好備用計劃。

A：是的，所以模型不必暫停並重新計算每次。

B：好的，這說得通。是的，現在說到效率，為了避免瓶頸，這避免了輔助損失的性能損失。

A：沒錯。它們還包括一個補充的序列平衡損失，是的，以防止個別過程中的極端不平衡。

B：是的，通過將每個標記限制為最多四個節點，它們減少了網絡通信。好的，這也有助於流暢化。

A：好的，讓我們來談談 DeepSeek V3 如何管理訓練的計算需求。我知道你對成本優化特別感興趣，以及他們如何以經濟方式做事。

B：是的，這個模型在這方面做了一些驚人的事情。

A：是的，平均每個標記選擇 3.2 個專家，這是一個很好的平衡來減少開銷。

B：沒錯。所以這是一個非常高效和有效的方法。

A：是的，這是一個非常聰明的方法，使這麼複雜的模型工作得如此出色。

B：是的，它們還通過這種方法實現了專家專業化。好的，這意味著不同的專家在不同的領域被激活。所以它們是什麼？

A：DeepSeek V3 使用 FPA 混合精度訓練框架。好的，這是一個對於這麼大規模的模型來說的重大突破。提醒我 FPA 是什麼？

B：當然，它是 8 位浮點數。

A：好的，它使用比傳統格式更少的位來表示數字。好的，這轉化為更少的記憶體和更快的計算。

B：沒錯。這就像壓縮一個大圖像文件，但你仍然得到圖像的精髓。它只是佔用更少的空間，對嗎？

A：沒錯。所以每個專家不是通用激活，而是在特定領域。所以它是精細調整並準備好行動。

B：是的。現在這種批次方式非常聰明。

A：是的，我同意。這種動態負載平衡方法非常有趣。這一切都是 DeepSeek V3 對性能和資源利用的承諾。

A：絕對。現在我們今天覆蓋了很多內容。這真的很有趣，但使用更少的位數可能會影響準確性嗎？

B：這是一個有效的擔憂，他們對此進行了仔細的處理。好的，他們實施了一些技術來減少任何潛在的準確性損失，包括精細量化。

A：是的，它允許對 FPA 中數字的表示方式進行精確控制。是的，從多頭潛在注意力到 DeepSeek Mo 和負載平衡，是的，這個 DeepSeek V3 模型是一個非常複雜的系統，這是創新推動我們…