**CH4 GDS庫和路徑查找**

1. **什麼是 Neo4j GDS庫？**

【定義】用於圖數據分析和圖算法的擴展庫，專為Neo4j圖數據庫而設計。它提供了一套豐富的圖算法和工具，用於在大規模圖數據集上進複雜的圖分析和數據挖掘任務。

【目標】提供高性能和可擴展的圖算法，以幫助用戶從複雜的圖數據中提取有價值的信息和見解。基於圖模型和圖算法理論，結合了圖數據庫的存儲和查詢優勢，能夠有效地處理大規模圖數據。

【其他】該庫提供了一系列常用的圖算法，例如最短路徑算法、社區發現算法、中心性算法、聚類算法等，以及更高級的算法，如圖嵌入和圖神經網絡。用戶可以使用Cypher查詢語言或專用的GDS庫函數來執行這些算法，並可以通過調整參數和配置來優化算法的性能和結果。

【應用】社交網絡分析、推薦系統、欺詐檢測、基因組學和生物信息學等領域。它為用戶提供了一個強大而靈活的工具集，幫助他們更好地理解和分析複雜的圖數據，從而做出更準確的決策和預測。

1. **Projected graph（投影圖）**

【定義】Projected graph 是一種基於原始圖數據的子圖，其中只包含特定類型的節點和關系。它是通過篩選和投影原始圖數據中的節點和關系而創建的。

【投影過程】通過執行 Cypher 查詢和使用 GDS 庫函數創建。用戶可以指定所需的節點類型和關系類型，以及其他條件來篩選和投影圖數據。

【用途】

提供靈活性：通過在投影圖上執行算法，可以減少數據量和複雜性，提高計算效率，並針對特定領域或問題進行更精確的分析。

加速計算：由於投影圖只包含特定類型的節點和關系，相比於原始圖數據，它的規模通常更小。這使得在投影圖上執行圖算法更高效，能夠更快地進行計算和分析。

靈活的數據探索：通過創建投影圖，用戶可以更方便地探索和理解與特定節點和關系相關的圖結構和模式。這有助於發現隱藏在大規模圖數據中的關聯性和見解。

|  |  |  |
| --- | --- | --- |
| **差異** | **Native Projection ( 原生投影 )** | **Cypher Projection** |
| 資料操作 | 使用圖資料庫的原生API進行資料操作 | 使用Cypher查詢語言進行資料操作 |
| 效率 | 高 | 低 |
| 精細度 | 低 | 高 |
| 適用 | 正式環境 | 開發環境時Debug、Test |

【種類】

1. **GDS 庫的常用算法 ( 只介紹最短路徑算法 )**

* 迪杰斯特拉（Dijkstra）最短路徑算法
* A\* ( A Star ) 最短路徑算法
* KSP ( K-Shortest Path )
* SSSP ( Singal Source Shortest Path )
* All-Pairs Shortest Path 全點對最短路徑算法

|  |  |
| --- | --- |
| 算法 | 用途 |
| 迪杰斯特拉 | 找到一個給定節點到其他所有節點的最短路徑 |
| A\* | 找到兩個節點之間的最短路徑，結合了啟發式函數和Dijkstra算法的思想 |
| KSP | 找到最短路徑中的前K個路徑，如導航系統中提供多個可選的路線 |
| SSSP | 找到給定節點到其他所有節點的最短路徑，可以並行，並行數由delta值決定。  當delta = 1時，SSSP≡迪杰斯特拉 |
| 全點對 | 計算任意兩個節點之間的最短路徑，適用於全局最短路徑計算 |

1. **圖優化**

【定義】在圖數據處理和圖查詢過程中，采用各種技術和方法來提高性能、效率和可擴展性的過程

【TSP旅行商問題】Traveling Salesman Problem

1. 問題：從起始節點開始，通過每個節點，且每條邊只能通過一次，最後繞回起始原點
2. 結論：這是一個NP-hard問題，意味著在一般情況下，找到最優解需要耗費大量的計算時間。且問題規模的指數級增長，求解TSP的確切解通常是不可行的，因此研究者們發展了各種啟發式算法和近似算法來尋找較優解。
3. 解法
   * ACO蟻群優化（Ant Colony Optimization）
   * GA基因演算法（Gene Alogorithm）
4. 擴展 🡪 MTSP（多重旅行商問題）

<eg.>快遞公司的「多台」貨車。 約束條件：車的容量、顧客指定交付時間

【生成樹】Spanning Tree

* 普里姆算法（Prim's algorithm）：用於求解最小生成樹（Minimum Spanning Tree，MST）的貪心算法。
  + 最小生成樹：連接一個連通無向圖的所有頂點，並使得樹上所有邊的權重之和最小。
  + 從一個頂點開始，逐步擴展最小生成樹的頂點集合，每次選擇一條連接已經包含的頂點和不包含的頂點的最小權重邊。通過不斷選擇權重最小的邊，直到最小生成樹包含了圖中的所有頂點。