連續性時序資料的隱私保護機制-以巨量個人定位資料為例

雲端運算暨多媒體實驗室

執行單位：國立嘉義大學/資訊管理系　主持人：林土量助理教授　計畫編號：MOST 105-2221-E-415-016

計畫摘要

本計畫主要探討如何匿名化連續性時序資料，避免個人隱私資料被惡意攻擊者利用序列資料探勘技術識別出來。匿名技術與資料探勘技術有其相關性，尤其連續性時間序列資料所隱含的知識通常都需要透過序列資料探勘技術來取得，因此在開發連續性序列資料的隱私保護機制時，必須把序列資料的探勘技術列入考量。

雲端的計算近年來受到極大的重視。許多的科學計算常需要巨額的軟硬體設備投資，因而讓研究人員望而怯步，雲端計算的發展可紓解研究人員在這方面的困擾，讓研究人員可以直接使用雲端計算資源供應商所提供之高效能與高穩定度的軟硬體平台，來快速獲取複雜科學計算的結果，以達成之前所未有的效率[1]。

本測試計畫對於系統的測試範圍包含單元測試、整合測試、系統測試，及接受度測試。單元測試即進行各函式的驗證及探討其結果，確認單一功能正常運作。接著整合測試驗證個單元之間是否能正常傳遞資料。最後是讓終端使用者進行系統測試以及接受度測試。

技術特色

(1)FP-Growth

FP-Growth演算法[2]則是利用FP-Tree資料結構將資料庫做壓縮，並將壓縮後的資料存放記憶體中，因此相較於Apriori演算法，在處理過程中掃描硬碟資料的次數僅只需兩次，且運算速度相對加快許多。FP-Tree資料結構能有效的壓縮資料庫。針對資料庫中的資料組合作最小支持度門檻值的篩選並建樹，以迭代的方式反覆對FP-Tree的路徑進行條件比對，並探勘出所有的頻繁項目集。

(2)MapReduce

雲端部分是透過MapReduce 運算架構，是由Google 所提出的一套軟體架構，是種分散式與平行化處理的程式設計模型，可以同時運行在多個不同電腦組成的叢集上，它主要是用來處理大量資料，擁有可靠的容錯機制。MapReduce[3, 4]的程式架構特性，能加快產生頻繁項目集的速度。

1.Map 階段：該函式所負責的任務是從主節點(master node)輸入的一個key/value 序對，將這組輸入分割成數個的子部分，分散到各個工作節點(worker nodes)上做運算。

2.Reduce 階段：該函式負責由主節點(master node)收回處理完的子部分，將子部分重新組合產生輸出。

(3)泛化保護

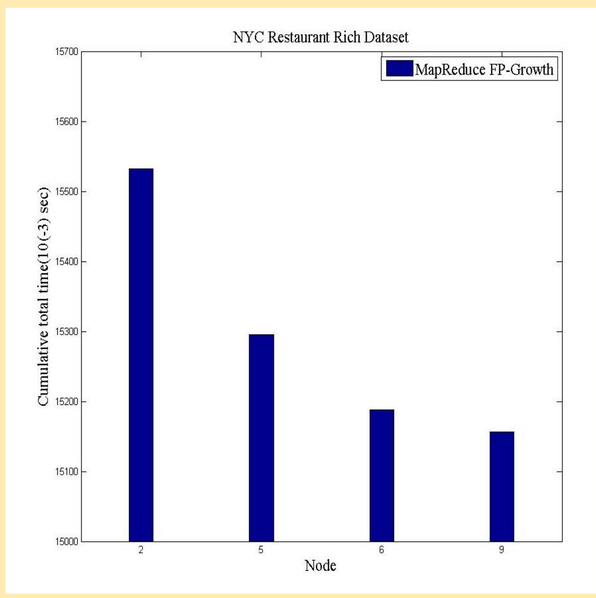
保護不頻繁景點則是透過泛化方式呈現，若從原始資料庫中透過資料探勘可以探勘出頻繁景點及不頻繁景點。假設透過探勘步驟可以得到頻繁景點為EA：5次，在不頻繁景點中探勘出EC：1次，我們團隊會結合頻繁景點和不頻繁景點，透過泛化的方式讓EA：5和EC：1轉換成EX：6。因此，在探勘階段，就不會單獨有一筆EC景點資料存在，透過泛化方式組合出EX：6，達到資料的隱私保護。

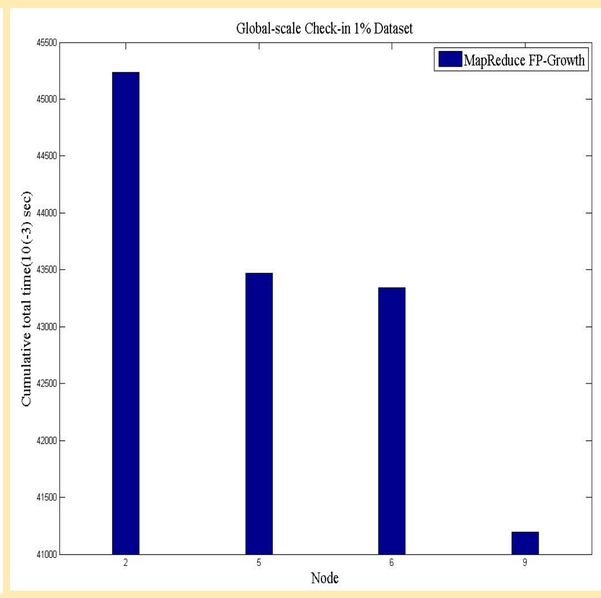
技術應用範圍

主要可應用於適地性服務LBS系統，經由隱私保護機制可以讓比較特殊的位置資訊做到相對應的保護，不會因為該位置較少人拜訪，就能透過關聯法則探勘出使用者的個資，因此本系統可達到隱私保護的目的。

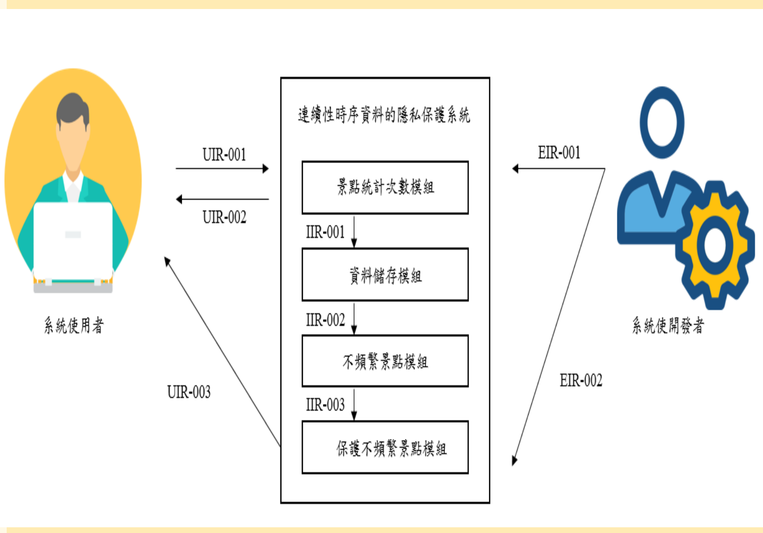
計畫結果

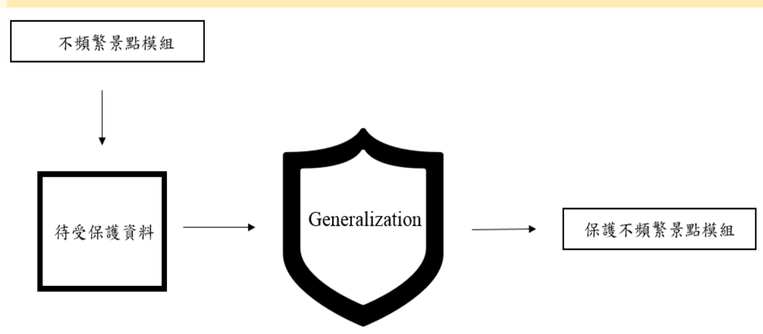
實驗部分利用了真實的打卡記錄做實驗，發現了透過雲端計算可以減少了運算時間。在比較過程中，利用了兩個不同的資料做測試，包含NYC Restaurant Rich Dataset 以及Global-scale Check-in 1% Dataset分別做實驗比較。資料型態包含了使用者以及打卡位置，實驗階段針對了兩台機器、五台、六台以及九台機器做比較。透過實驗數據可得知，只要加入雲端架構的電腦數量越多，則運算速度可以提升些許的差異。下圖為加入的電腦數以及速度的比較做長條圖比對。





計畫架構





本計畫的架構由景點統計次數模組、資料儲存模組、不頻繁景點模組，以及保護不頻繁景點模組要針對不常出現的場所做保護。系統中的模組之間都需要進行整合測試，各項參數與數據必須能夠在資料流正常傳遞，並檢視測試結果是否能符合本計畫制定的接受準則。系統測試環境架構圖如左所示。

參考資料

1. V. Garg, S. Arora, and C. Gupta, “Cloud computing approaches to accelerate drug discovery value chain,” Combinatorial chemistry & high throughput screening, vol. 14, no. 10, pp. 861-871, 2011.
2. J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation." pp. 1-12.
3. X. Lin, "Mr-apriori: Association rules algorithm based on mapreduce." pp. 141-144.
4. L. Chunqing, “Apriori Algorithm Optimization Study Based on MapReduce,” 2015.