

分析與衡量撰寫結構化查詢語言敘述的任務複雜度

許峻維¹ 李之中² 施教旺³ 鍾遜彰⁴

¹ 中華大學資訊管理學系 M09710008@chu.edu.tw

² 中華大學資訊管理學系 leecc@chu.edu.tw

³ 龍華科技大學資訊管理學系 joseph@mail.lhu.edu.tw

⁴ 中華大學資訊管理學系 M09810017@chu.edu.tw

摘要

在資料庫普遍於資訊業界使用的今日，如何衡量使用者撰寫 SQL 敘述的任務複雜度，並藉此評估使用者撰寫 SQL 敘述的績效是一個值得探討的議題。目前 Halstead 方法與運算子權重法是兩個僅有衡量撰寫 SQL 敘述的任務複雜度的方法。但是，Halstead 方法與運算子權重法所衡量的撰寫 SQL 敘述的任務複雜度，與使用者實際撰寫 SQL 敘述的感受並不相同。本研究的目的為設計一個能夠貼近使用者撰寫 SQL 敘述任務複雜度的新衡量方法—延伸式 Halstead 方法。並透過實驗說明延伸式 Halstead 方法所衡量的使用者撰寫 SQL 敘述任務複雜度相對於使用 Halstead 方法與運算子權重法所衡量的使用者撰寫 SQL 敘述任務複雜度更接近使用者實際撰寫 SQL 敘述所感受的任務複雜度。在實驗中，本研究透過四個題組測驗取得使用者的作答時間、信心度與正確性，並據此作為分析 Halstead 方法、運算子權重法與延伸式 Halstead 方法三種衡量方法優劣。在資料分析中，本研究以題組內分析討論運算元對於撰寫 SQL 敘述任務複雜度的影響，以題組間分析討論運算子對於撰寫 SQL 敘述任務複雜度的影響，以順序維度分析衡量三種方法的數值序列與使用者感受的一致程度。結果顯示延伸式 Halstead 方法因同時考量運算子與運算元的影響，因此相較於 Halstead 方法與運算子權重法，其所衡量的使用者撰寫 SQL 敘述的任務複雜度，最貼近使用者實際之感受。

關鍵詞：結構化查詢語言、任務複雜度、Halstead 方法、運算子權重法、順序維度。

分析與衡量撰寫結構化查詢語言敘述的任務複雜度

1. 緒論

資料庫和資料庫系統在現今社會中的應用非常廣泛，在不同類型的公司中幾乎都可以看到資料庫應用系統的使用，例如，在網路上購買各類商品，或是目前流行的各類社交網站，都是資料庫應用系統的運用的例子。資料庫應用系統需要由龐大的資料庫作為後端支援前端網站的各項功能。上述的資料庫應用系統中，前端網站向後端資料庫擷取資料時，需要向後端資料庫發出結構化查詢語言(Structured Query Language, SQL)敘述，根據該 SQL 敘述自資料庫中擷取資料再交給前端。而前述資料庫應用系統中的 SQL 敘述需要資料庫應用系統開發者於資料庫應用系統開發時完成撰寫並嵌入於資料庫應用系統中。因此如何評估資料庫應用系統開發者撰寫 SQL 敘述的績效成為一個重要的議題。

在學者 Chan 等(1999)結合學者 Reisner(1981)與學者 Chan 等(1993)所進行的研究中指出影響使用者查詢績效(Query performance)的因子應包含資料表示式(Data expression)、任務複雜度(Task complexity)、查詢表達方式(Query expression)、使用者特性(User characteristics)與系統特性(System characteristics)等五個因子。上述五個因子中的任務複雜度即為撰寫 SQL 敘述的任務複雜度。在這些使用者查詢績效領域的論文中，就本研究能力範圍所及的調查僅發現有少數研究探討任務複雜度對使用者查詢績效的影響。然而在這些研究中(Bowen, 2009)(Liu, 2011)(Topi, 2005)，學者僅將任務複雜度簡單的二分為「簡單」與「複雜」兩種類別，並據以分析不同任務複雜度對使用者查詢績效的影響。也因此不同學者對於區分撰寫 SQL 敘述任務複雜度為「簡單」或「複雜」的作法各不相同，莫衷一是。因此學界需要一個具體衡量撰寫 SQL 敘述任務複雜度的方法。

在探討衡量撰寫 SQL 敘述任務複雜度的研究中，本研究發現兩個針對衡量撰寫 SQL 敘述任務複雜度進行討論的研究工作。一是學者 Borthick 等(2001)的研究中，提出使用 Halstead 方法來衡量撰寫 SQL 敘述的任務複雜度。二是學者袁輝偉(2010)提出的運算子權重法。前者的優點為可以簡單及客觀的衡量出 SQL 敘述任務複雜度，但是在前者的方法中，對於所有的運算子都是同地位的，因此只要運算元越多，衡量出的數值就會相對的上升，會產生 SQL 敘述越長複雜度就越高的情況。而後者為了改進前者方法的缺點，提出應給予 SQL 敘述中運算子不同的權重後，根據運算子的權重來決定查詢命令的任務複雜度，但卻忽略了運算元對 SQL 敘述任務複雜度的影響。以上兩個研究工作的衡量撰寫 SQL 敘述任務複雜度的方式，其衡量出來的結果，與使用者實際撰寫 SQL 敘述所感受的任務複雜度並不相同。本研究希望找出一個能夠更貼近使用者實際撰寫 SQL 敘述所感受的任務複雜度的衡量方法。為達到上述的研究目的，本研究將 SQL 查詢敘述分為運算子與運算元兩個部分，運算子為 SQL 敘述各種不同的 SQL 關鍵字組合做為運算子，而各種不同 SQL 敘述所需的表格、欄位或是條件式做為運算元，依照這樣的規則設計測驗問題，並以不同的衡量方法算出各自所代表的數值順序，並以此排序

與實際使用者查詢績效的結果排序後做比較。

為了衡量使用者撰寫 SQL 敘述任務複雜度，本研究設計延伸式 Halstead 方法 (Enhanced Halstead)，並使用這個方法衡量使用者撰寫 SQL 敘述任務複雜度。本研究並透過實驗說明使用延伸式 Halstead 方法所衡量的使用者撰寫 SQL 敘述任務複雜度相對於使用 Halstead 方法與運算子權重法所衡量的使用者撰寫 SQL 敘述任務複雜度更接近使用者實際撰寫 SQL 敘述所感受的任務複雜度。在實驗中，本研究透過四個題組測驗取得使用者的作答時間、信心度與正確性，並據此作為分析 Halstead 方法、運算子權重法與延伸式 Halstead 方法三種衡量方法優劣。在資料分析中，本研究以題組內分析討論運算元對於撰寫 SQL 敘述任務複雜度的影響，以題組間分析討論運算子對於撰寫 SQL 敘述任務複雜度的影響，以順序維度分析衡量三種方法的數值序列與使用者感受的一致程度。結果顯示延伸式 Halstead 方法因同時考量運算子與運算元的影響，因此相較於 Halstead 方法與運算子權重法，其所衡量的使用者撰寫 SQL 敘述的任務複雜度，最貼近使用者實際之感受。

本論文後續論文如下。第二節文獻探討，依序介紹結構化查詢語言、任務複雜度、以及當前有關結構化查詢語言與任務複雜度之間關係的相關研究與衡量結構化查詢語言任務複雜度之方法。第三節說明本研究之研究設計與參數的定義、題目設計、衡量方法介紹及研究對象。各題組測驗之後的結果與所有測驗資料的結果應用不同的衡量方法衡量結構化查詢語言任務複雜度的結果做分析與討論則在第四節中說明。第五節歸納本研究資料分析後的結論與說明研究限制與未來研究之建議。

2. 文獻探討

本節針對本研究的相關文獻進行討論。在 2.1 節中簡介 SQL 的查詢敘述，2.2 節介紹任務複雜度，2.3 節則介紹目前衡量撰寫 SQL 敘述任務複雜度的 Halstead 與運算子權重兩個方法。

2-1 SQL 的查詢敘述

在資料庫的應用中，使用者使用 SQL 中的 SELECT 敘述自資料庫中取出所需的資料。SELECT 敘述基本語法如下：

```
SELECT <attribute list>  
FROM <table list>  
WHERE <condition>
```

其中 <attribute list> 為查詢參考的屬性名稱，<table list> 為查詢參考的表格名稱，<condition> 為查詢參考的條件式。

除了上述的基本語法外，在 SQL 還有很多較為複雜的 SELECT 敘述語法。例如，進行表格合併 (JOIN) 與運用子查詢 (SUB QUERY) 的 SELECT 敘述。SELECT 敘述中進行表格合併的語法為

```
SELECT <attribute list>
```

FROM <table list> JOIN <table list> USING(<attribute list>)

關鍵字 JOIN 主要的功能是在做表格的合併，使用者可以利用關鍵字 USING 後所指定的屬性集進行表格合併，在取得所需要的資料。而在 SELECT 敘述中使用子查詢的語法為

SELECT <attribute list>

FROM <table list>

WHERE <condition> <operator> (<SUB QUERY>)

其中 <SUB QUERY> 查詢時所參考的子查詢。

2-2 任務複雜度

任務複雜度可以分為三個組成的要素，結果(Product)、動作(Acts)及資訊線索(Information cues)(Wood, 1986)。結果是個體執行行為或動作後所預期產生的結果；動作是有特定目的的行為模式，例如：聽、說、讀、寫等；資訊線索則是個人在執行這些特定的行為模式時所需要使用到的資訊。

任務複雜度大致可分為兩種(Maynard, 1997)，主觀任務複雜度(Subjective complexity)與客觀任務複雜度(Objective complexity)，兩者最大的不同在於，前者較易受到個人經驗及能力的影響，因此不同的人對同樣的任務可能會感受到不一樣的主觀任務複雜度，而後者則較不易受影響(Campbell, 1988)；本研究期望找出可以有效衡量客觀任務複雜度的方法，因此本研究分別找三群不同程度的受測者接受測驗，依照測驗後的結果排列順序後，找出衡量方法中最符合實際排列順序的方法。

對於何謂複雜的任務，根據 Wood(1986)的定義為，越是複雜的任務，通常需要越多的資訊線索才能夠進行任務的執行，但當資訊線索的量超過個人所能處理的上限，會發生「超載」(Overload)，而超載將會導致較差的任務績效，因此越是複雜的任務，任務績效也會越差，本研究將根據此定義來設計測驗題目。

根據 Wood(1986)對於任務複雜度的定義，越複雜的任務將導致越差的任務績效，將此定義對照到 SQL 查詢敘述的研究上，也就是越複雜的 SQL 查詢敘述將導致越差的使用者查詢績效，學者 Chan 等(1999)結合學者 Reisner(1981)與學者 Chan 等(1993)所進行的研究中指出使用者查詢績效(Query performance)應包含時間(Time)、信心度(Confidence)與正確性(Accuracy)。

2-3 撰寫結構化查詢語言之任務複雜度

本節將介紹有關結構化查詢語言任務複雜度之相關研究，Borthick 等(2001)所提出的 Halstead 方法與學者袁輝偉(2010)所提出的運算子權重法是當前衡量撰寫 SQL 敘述任務複雜度的兩個具體方法，本節將對這兩個方法進行介紹。

2-3-1 Halstead 方法

在 Borthick 等(2001)所提出的以 Halstead 方法中，將 SQL 敘述中的字符(Token)區分為運算子(Operators)與運算元(Operands)，並以下列式子得到撰寫該 SQL 敘述的任務複雜度(Halstead, 1977)。該式如下所式：

$$D = (N \log_2 N) / (n^* \log_2 n^*) \tag{1}$$

其中 $N = N_1 + N_2$ ， N_1 為查詢敘述中運算子的個數， N_2 為查詢敘述中運算元的個數； $n = n_1 + n_2$ ， n_1 為查詢敘述中相異運算子的個數， n_2 為查詢敘述中相異運算元的個數； n^* 為最短查詢敘述的 n 其值為 5(因為最短的查詢敘述為 SELECT * FROM table)。

在此我們舉一個例子說明如何使用 Halstead 方法衡量撰寫以下 SQL 敘述的任務複雜度。如果有一 SQL 敘述如下：

```
SELECT E_NAME, CITY
FROM (EMPLOYEE JOIN DEPARTMENT USING (D_ID)) JOIN
LOCATIONS USING (LOCATION_ID),
```

該查詢敘述為找出所有員工的姓名(E_NAME)與工作部門位於的城市(CITY)，以 Halstead 方法計算其查詢命令任務複雜度，我們先將 SQL 敘述中的運算子與運算元分類如圖 2-1。

運算子	數量	運算元	數量
SELECT	1	E_NAME	1
,	1	CITY	1
FROM	1	EMPLOYEE	1
()	3	DEPARTMENT	1
JOIN	2	D_ID	1
USING	2	LOCATIONS	1
;	1	LOCATION_ID	1
$n_1 = 7$		$n_2 = 7$	
$N_1 = 11$		$N_2 = 7$	

$$D = (18 \log_2 14) / (5 \log_2 5) = 5.90$$

圖 2-1 一個使用 Halstead 方法計算查詢敘述任務複雜度的例子

Halstead 方法同時考慮運算子與運算元的衡量方法，合乎目前文獻對於任務複雜度的定義，但是本研究對於 Halstead 方法將運算子與運算元視為同等的並且所有的運算子都是相同的觀點並不認同，而且 Halstead 方法在 SQL 查詢敘述字數增加時，複雜度就會相對的上升，在實際測驗後是不符合實際情況的，詳細的分析將在第 4 節中進行。

2-3-2 運算子權重法

學者袁輝偉(2010)提出的運算子權重法，主要是利用 SQL 查詢敘述中運算子的權重來衡量 SQL 查詢敘述的任務複雜度，學者袁輝偉設計了 24 題題目來給受測者做測驗後，計算所有受測者的答題正確率，得到運算子複雜度量表，如表 2-1。

表 2-1 SQL 查詢敘述運算子複雜度量表

運算子	SELECT	WHERE	ALIAS	DISTINCT	BETWEEN	HAVING
複雜度	1	10	23	21	28	16
運算子	IN	LIKE	ORDER BY	JOIN	GROUP BY	SUB QUERY
複雜度	43	43	26	17	56	34

我們利用這個量表計算撰寫下列 SQL 查詢敘述的任務複雜度。

```
SELECT E_NAME, CITY
FROM (EMPLOYEE JOIN DEPARTMENT USING (D_ID))
JOIN LOCATIONS USING (LOCATION_ID)
```

以運算子權重法計算，因 SQL 查詢敘述中出現 SELECT 及 JOIN 兩個運算子，量表中 SELECT 的複雜度為 1，JOIN 的複雜度為 17，而 JOIN 在查詢命令中出現兩次，所以本題答案以運算子權重法計算的複雜度為 1(SELECT)+17(JOIN)+17(JOIN)等於 35。

運算子權重法雖然可以根據量表很簡單的計算出 SQL 查詢敘述任務複雜度，但僅考慮運算子權重的作法，與相關文獻中對於任務複雜度的定義並不相符，而且當兩個查詢敘述其運算子的使用完全相同，而僅有運算元的差異時，運算子權重法無法衡量兩者之間的差別，此觀點將在第 4 節資料分析中討論詳細解說。

3. 研究方法

在本節中將介紹本論文的研究方法，在 3.1 節介紹研究架構，3.2 節介紹題目設計，3.3 節講述預測、實施測驗與批改答案的方法，3.4 節介紹本研究所提出的 Enhanced Halstead 衡量複雜度方法，3.5 節說明本研究的研究對象，3.6 節則介紹本研究資料分析的過程。

3-1 研究架構

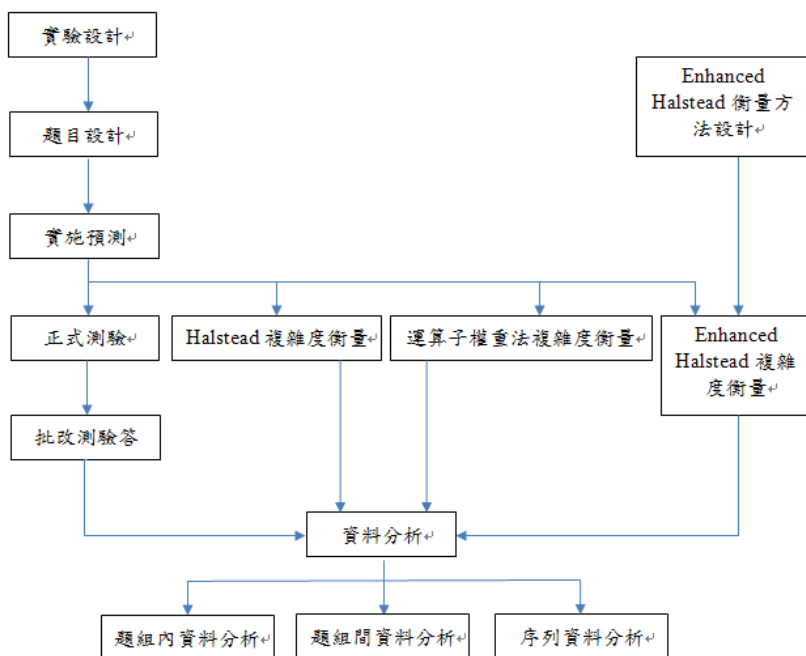


圖 3-1 研究架構圖

本研究的研究架構如圖 3-1 所示。在這個研究架構中本研究的首要工作為「Enhanced Halstead 衡量方法的設計」。本研究認為撰寫 SQL 敘述的任務複雜度，應該由 SQL 敘述中的運算子與運算元共同決定，同時不同運算子的複雜度也不同。因此本研究以 Halstead

方法與運算子權重法為基礎，提出一個由所撰寫 SQL 敘述中個別運算子複雜度與運算元數量共同衡量任務複雜度的方法。本研究稱這個方法為延伸式 Halstead 方法(Enhanced Halstead)。有關延伸式 Halstead 方法的更深入說明，將在 3.4 節中進行。

在完成架構中「Enhanced Halstead 衡量方法設計」後，本研究透過實驗說明延伸式 Halstead 方法相較於 Halstead 方法與運算子權重法，更能貼近使用者實際撰寫 SQL 敘述所感受的任務複雜度。在實驗設計中，首先取得使用者實際撰寫 SQL 敘述的任務複雜度。但是任務複雜度在實務上難以衡量，因此本研究透過使用者撰寫 SQL 敘述的績效衡量任務複雜度。本研究使用撰寫 SQL 敘述所使用的時間、信心度和正確性(Borthick(2001)，袁輝偉(2010))衡量使用者撰寫 SQL 敘述的績效。為了具體衡量上述三個變數，本研究將正確性轉換為受測者作答結束後的分數來代表，而使用時間為受測者於實驗中完成 SQL 敘述所使用的時間，至於信心度則以在受測者於完成作答之後自我評估是否答對問題的程度，上述三個變數在本研究中均為撰寫 SQL 敘述任務複雜度的操作型變數。

完成實驗設計之後，本研究接著進行「題目設計」進行實驗題目的設計，本實驗題目設計的內容，將在 3.2 節中說明。「題目設計」完成之後接著進行「預測實施」以找出實驗問題可能的不足之處。在完成「預測實施」之後，進行「正式測驗」，於實驗中取得使用者撰寫 SQL 敘述的時間與信心度。而正確率則以 Borthick 等學者(2001)所建議的方法批改受測者答案後取得。在完成上述工作之後，本研究將各測驗題目題號依據實驗中所取得各個題目的答題時間由低而高排列，而信心度與正確率，則由高而低排列。因此本研究得到三個序列，分別稱之為時間基準序列、信心度基準序列與正確率基準序列。這三個基準序列在本研究中將用來表示使用者實際撰寫 SQL 敘述所感受的任務複雜度。

接著本研究分別使用 Halstead 方法、運算子權重法以及本研究提出的 Enhanced Halstead 方法取得這三個方法在實驗中各個題目的任務複雜度，並將使用同一方法衡量實驗題目所取得的任務複雜度，並依照所衡量得到任務複雜度由低而高排列題號。因此本研究可以得到三個序列，其中使用使用 Halstead 方法進行任務複雜度衡量所取得的序列稱為 Halstead 序列，使用運算子權重方法進行任務複雜度衡量所取得的序列稱為運算子權重序列，而使用使用 Enhanced Halstead 方法進行任務複雜度衡量所取得的序列稱為 Enhanced Halstead 序列。

3-2 題目設計

本研究參考 Borthick 等(2001)對於 SQL 查詢敘述的定義，將 SQL 查詢敘述分為運算子及運算元兩部分，實驗題目則根據上述的分類進行設計。首先將實驗題目分成四個題組，每個題組包含若干個題目，在同屬一個題組中的題目間，改變選定運算子的運算元個數。題組的名稱使用被改變運算元個數的運算子名稱命名。在本實驗題目中選擇了 SELECT、WHERE、JOIN 及 SUB QUERY 等四個運算子，分別探討這四個運算子它們的運算元數量改變時撰寫這些 SQL 敘述工作複雜度的變化情形。因此本實驗題目中包含了四個題組，分別為 SELECT、WHERE、JOIN 及 SUB QUERY 等四個題組。

3-3 預測、實施測驗與批改方法

在完成實驗題目設計之後，接下來對實驗題目進行預測。預測所使用的實驗環境與後續正式實驗的環境完全相同。預測由三位中華大學碩士班的同學實施，這三位同學至少曾修習三門資料庫課程。

正式實驗時，先請受測者註冊自己的帳號密碼，以便網站資料庫能夠據此紀錄受測者答題資料。接著測試者使用剛註冊的帳號登入測驗資訊系統，此時資訊系統同時呈現作答所需的實體關係模型(E-R Model)以及與實體關係模型相應的關聯式資料模型(Relational Model)給受測者參考。在實驗進行中，受測者依題號循序進行作答，測驗資訊系統在受測者作答時將同時紀錄受測者的作答時間，完成作答後由系統紀錄該題答案並進行信心度的詢問，信心度代表受測者對於本題答案正確回答問題的信心，信心度計分方式為1~5，分數越高，代表受測者對於本題答案正確的信心越強；受測者所有答題結束後，網站將自動登出受測者的帳號，並無法再次登入，避免受測者重複修改答案影響資料的真實性。

上述答題時間與信心度於完成實驗實施之後即可自測試資訊系統得知，正確率則需批改受測者的答案後方能獲得。本研究以 Borthick 等學者(2001)建議的方法進行受測者測試答案的批改。Borthick 等學者(2001)將受測者所回答的答案可能發生的錯誤分為兩種—微觀錯誤(Micro Errors)與巨觀錯誤(Macro Errors)。微觀錯誤指的是可以經過一次修改就成為正確查詢的錯誤，而巨觀錯誤指的是會影響到查詢結果欄、行或者是集合的錯誤。Borthick 等(2001)的研究經由計算這兩種錯誤出現的次數來當作使用者查詢績效的正確性，但本研究並不去計算微觀錯誤與巨觀錯誤的數量，而是以錯誤計分，每題滿分為10分每出現一個微觀錯誤扣2.5分，一個巨觀錯誤則扣10分，以此方式計分，這樣的方式可以將微觀錯誤與巨觀錯誤具體化來比較，而不用將微觀錯誤與巨觀錯誤的數量分開來檢視，讓使用者查詢績效正確性的表現更為明確。

3-4 延伸式 Halstead 方法

本研究從文獻調查進行衡量撰寫 SQL 敘述任務複雜度學理探討，在這個學理探討中，本研究發現 Borthick 等學者(2001)所使用的 Halstead 方法與學者袁輝偉(2010)所提出的運算子權重法，在方法設計上均有未臻周延的地方。因此導致在衡量撰寫 SQL 敘述任務複雜度時，Halstead 方法與運算子權重法均無法正確衡量撰寫 SQL 敘述的任務複雜度。本研究發現撰寫 SQL 敘述的任務複雜度，應該由 SQL 敘述的運算子與運算元(包含屬性、表格以及的條件式)共同決定，同時不同運算子之間所具有的任務複雜度亦有不同。在 Halstead 方法中，Halstead 方法同時使用了 SQL 敘述的運算子與運算元衡量 SQL 敘述的任務複雜度。在實際的衡量方法上 Halstead 方法透過 SQL 敘述中的運算子的與運算元的數量衡量撰寫 SQL 敘述的任務複雜度。基本上 Halstead 方法認為撰寫 SQL 敘述的任務複雜度與所撰寫 SQL 敘述的長度(運算子+運算元的個數)成正比。因此兩個敘述，如果其中一個包含 SELECT、FROM、WHERE 的 SQL 敘述，與另一個包含合併運算的 SQL 敘述，如果兩個 SQL 敘述長度相同的話，這兩個 SQL 敘述的任務複雜度幾近相同。就資料庫系統研究社群的一般認知，撰寫一個合併運算的 SQL 敘述其任務複

雜度應該比撰寫一個包含 SELECT、FROM、WHERE SQL 敘述的任務複雜度來的高。因此，學者袁輝偉(2010)提出的運算子權重法衡量撰寫 SQL 敘述的任務複雜度。運算子權重法針對不同的 SQL 運算子訂定了不同的權重，藉此處理 Halstead 方法中的不足之處，但是運算子權重法僅考慮應該給運算子不同的權重，並未考慮運算元也同時影響撰寫 SQL 敘述時的任務複雜度。

本研究參考 Halstead 與運算子權重兩個衡量撰寫 SQL 查詢敘述任務複雜度的方法後，認為完整有效的衡量 SQL 查詢命令任務複雜度的方法應該同時具備運算子的權重以及運算元的數量兩個因素，因此將參考運算子加權法與 Halstead 方法，將兩個方法結合為延伸式 Halstead 方法(Enhanced Halstead)。

延伸式 Halstead 方法利用運算子權重法的運算子量表來當作 SQL 查詢敘述中運算子的權重，並在 Halstead 方法公式變型後加入運算子的權重，如下所示：

$$D = \frac{(n_1 + \alpha N_1) \log(n_2 + N_2)}{(n^* \log n^*)} \quad (2)$$

其中 N_1 為查詢敘述中運算子的個數； N_2 為查詢敘述中運算元的個數； n_1 為查詢敘述中相異運算子的個數； n_2 為查詢敘述中相異運算元的個數； α 為運算子權重。

本研究將 Halstead 方法中有關運算子的變數相加後並給予變數 N_1 運算子的權重，因為 N_1 為查詢敘述中運算子的個數，將運算子的權重放在這裡，會使運算子權重產生較明顯的作用，而又不會對運算元影響太顯著，而變成完全注重運算子的衡量方法。運算子權重部分，本研究同樣參考袁輝偉(2010)運算子權重法的運算子權重量表，但因為量表中學者袁輝偉的運算子權重定義數值太過龐大，如果直接使用的話，無法與其他衡量方法做適當的比較，因此本研究將運算子權重量表數值縮減為 1/10，表 3-1 為本研究所使用的運算子權重。

表 3-1 SQL 查詢敘述運算子複雜度量表

Operators	SELECT	WHERE	JOIN
Complexity	0.1	1	1.7

子查詢的部分不再使用學者袁輝偉所定義的權重去計算，因為子查詢的寫法有很多種，如果只固定的使用同一種運算子權重，無法客觀有效的衡量使用者的任務複雜度感受，而本研究子查詢使用的是 WHERE 命令，因此在權重部分使用的是 WHERE 的權重。

3-5 研究對象

本研究是探究衡量 SQL 語法複雜度方法的正確性，因此受測者必須接受過撰寫 SQL 語法的訓練，並且了解實體關係模型與關聯式資料模型。本研究共找了 100 位接受過撰寫 SQL 語法的受測者進行測驗，其中共有 19 位中華大學碩士專班學生(以下簡稱群組 B)接受過撰寫 SQL 語法訓練或工作上可能有實際操作經驗，其餘 81 位則為大學部學生，其中 26 位是國立台灣科技大學學生(以下簡稱群組 A)是受過兩學期撰寫 SQL 語法訓練課程的學生，另外 55 位則為中華大學二年級學生(以下簡稱群組 C)是受過一學期 SQL 語法訓練的學生。

3-6 資料分析

本研究所進行的資料分析包含「題組內資料分析」、「題組間資料分析」以及「序列資料分析」。在題組內資料分析中，探討同一運算子在運算元數量變化時對任務複雜度的影響，在題組間資料分析中，探討不同運算子在運算元數量固定時對任務複雜度的影響。根據研究題目設計，題目共分為 4 個題組，各題組內運算子不變，只做運算元數量的變化，題組內分析主要就是在觀察固定運算子的情況下，運算元數量對於 SQL 查詢命令任務複雜度的影響；題組間資料分析，則是選擇不同題組間，使用相同數量運算元的題目來做分析，觀察固定運算元數量的情況下，運算子對於 SQL 查詢命令任務複雜度的影響。序列資料分析則是利用整理完受測者作答資料後所產生的時間基本序列、信心度基本序列與正確率基本序列與 Halstead 序列、運算子權重序列與 Enhanced Halstead 序列做比較分析。在序列資料分析中使用順序理論(Order theory)定義順序維度(Order degree)進行序列資料分析，順序維度將在下一小節進行介紹。

3-6-1 順序維度

為了定義順序維度，本論文首先定義基準序列 S_q 為一個偏序集合 (Q^+, \leq_q) (Partial ordered set)， Q^+ 為基準序列 S_q 中所有題目題號的集合，而 \leq_q 是一個在 Q^+ 上定義的二元關連(binary relation)。對於任意兩個在 Q^+ 中題目的題號，在 \leq_q 中的值組 (q_i, q_j) 表示在基準序列 S_q 中題目 q_i 在題目 q_j 之前。接著本研究定義 Halstead 序列、運算子權重序列以及延伸式 Halstead 序列為測試序列。令測試序列 S_t 為一個偏序集合 (Q^+, \leq_t) ，其中 Q^+ 為測試序列 S_t 中所有題目題號的集合，而 \leq_t 是一個在 Q^+ 上定義的二元關連。對於任意兩個在 Q^+ 中題目的題號，在 \leq_t 中的值組 (t_i, t_j) 表示在測試序列 S_t 中題目 t_i 在題目 t_j 之前。根據以上的定義，本研究定義順序維度 R 為

$$R = \frac{|\leq_q - \leq_t|}{|\leq_q|} \quad (3)$$

例子：假設本研究使用三個撰寫 SQL 敘述的題目(題號分別為 1, 2, 3)進行實驗，在實驗完成之後發現受測者回答此三個問題的依其正確率高低的題號排列的基準序列 S_q 為 $q_1 \leq q_2 \leq q_3$ 。此三個撰寫 SQL 敘述的題目使用 Halstead 方法進行複雜度衡量依其獲得的任務複雜度高低的題號排列的測試序列為 $q_2 \leq q_1 \leq q_3$ 。因此，基準序列 S_q 的偏序集合為 (Q^+, \leq_q) ，其中 Q^+ 為 $\{q_1, q_2, q_3\}$ ， \leq_q 為 $\{(q_1, q_2), (q_1, q_3), (q_2, q_3)\}$ 。而測試序列 S_t 的偏序集合為 (Q^+, \leq_t) ，其中 Q^+ 為 $\{q_2, q_1, q_3\}$ ， \leq_t 為 $\{(q_2, q_1), (q_1, q_3), (q_2, q_3)\}$ 。根據上述的定義 $|\leq_q - \leq_t| = |\{(q_1, q_2), (q_1, q_3), (q_2, q_3)\} - \{(q_2, q_1), (q_1, q_3), (q_2, q_3)\}| = |\{(q_1, q_2)\}| = 1$ 且 $|\leq_q| = |\{(q_1, q_2), (q_1, q_3), (q_2, q_3)\}| = 3$ 。則順序維度 R 為：

$$R = \frac{|\leq_q - \leq_t|}{|\leq_q|} = \frac{1}{3} = 0.33$$

因此當順序維度 R 值越大時，代表兩個序列間排序的相異的排序越多，順序維度 R 值越小則代表兩個排序間相異的排序越少。

4. 資料分析結果與討論

本研究使用實驗所取得的資料進行資料分析。資料分析包含題組資料分析、題組間資料分析以及使用順序維度進行的順序分析。同題組資料分析探討 SQL 語法中不同數量的運算元在相同運算子時任務複雜度是否有所差別。題組間資料分析利用不同題組間相同運算元數量的題目探討 SQL 語法中不同運算子間任務複雜度的關係，最後利用順序維度進行的順序維度分析，分析運算子權重法、Halstead 及延伸 Halstead 三個方法與實際資料的排序差異程度，藉此找出哪個方法可以有效的衡量撰寫 SQL 敘述的任務複雜度。

4-1 題組資料分析

利用題組設計，分析同題組中相同運算子搭配不同數量運算元時，受測者在使用者績效、時間及信心度的變化，這樣的方式可以詳細的解釋 SQL 語法中不同數量的運算元在相同運算子時任務複雜度的變化。

4-1-1 題組內平均分數分析

由圖 4-1 可以看到在 SELECT 題組中使用者對於不同運算元數目時平均分數的變動。基於任務複雜度的學理，使用者在運算元數目越多時，平均分數應該要越低，因為 SQL 語法的任務複雜度提升，使用者的查詢績效應下降。但是此一現象在 SELECT 題組時表現並不明顯，然而在 WHERE 題組(圖 4-2)、JOIN 題組(圖 4-3)及 SUB QUERY 題組(圖 4-4)中都可以明顯的觀察到隨著運算元的數量增加，使用者的查詢績效越來越低，可以證實查詢敘述中運算元的多寡，對使用者的查詢績效有著一定程度的影響。因此，運算子權重法衡量 SQL 查詢複雜度方法中，只考慮 SQL 語法中運算子的衡量方法需要進一步的考量。

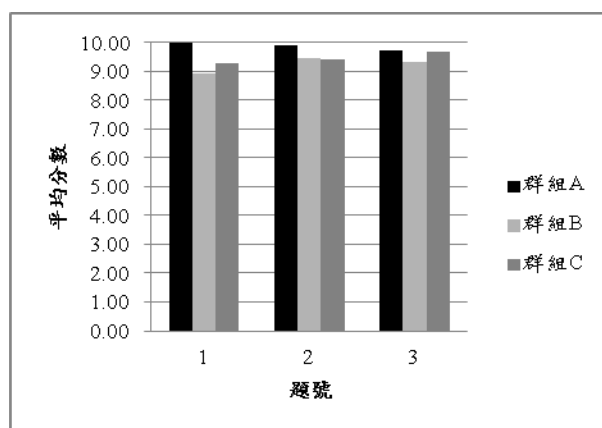


圖 4-1 SELECT 題組平均分數

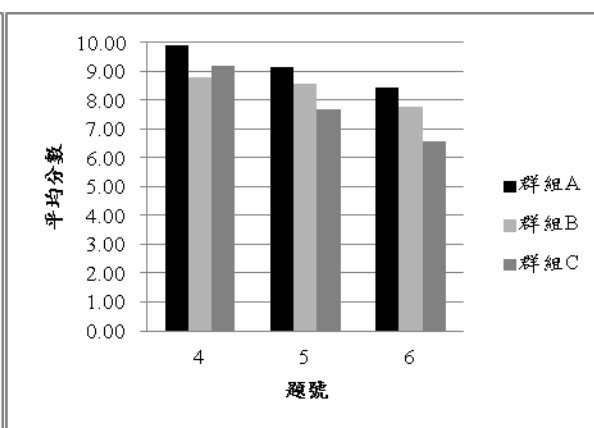


圖 4-2 WHERE 題組平均分數

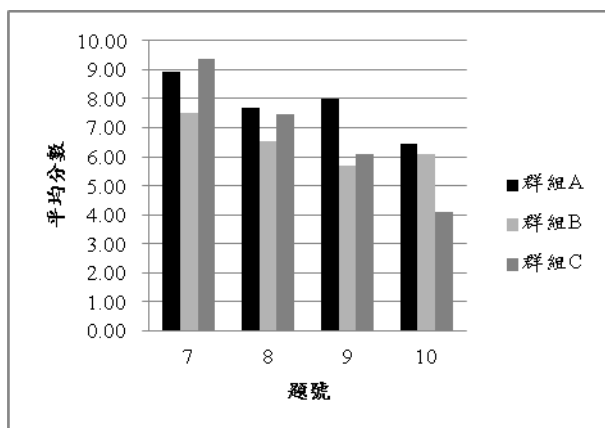


圖 4-3 JOIN 題組平均分數

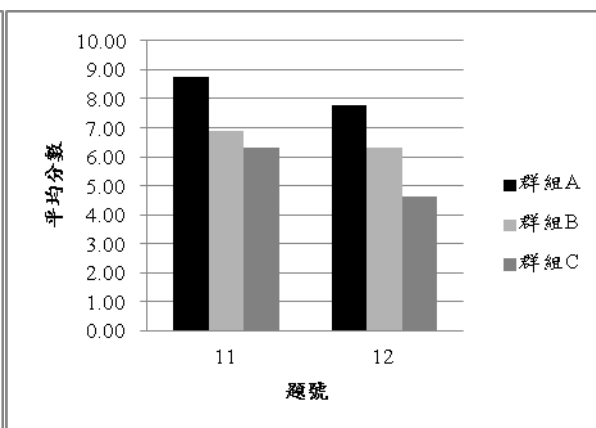


圖 4-4 SUB QUERY 題組平均分數

4-1-2 題組內平均作答時間分析

在本研究的實驗中，受測者登入測驗網站作答時，測驗系統將自動紀錄作答時間。在實驗完成之後，測驗系統也自動計算每位受測者的平均作答時間。以下將三個不同群組的受測者的作答時間資料整理後進行分析。在 SELECT 題組(圖 4-5)、WHERE 題組(圖 4-6)及 SUB QUERY 題組(圖 4-8)皆可以看到群組 A(台科大)與群組 C(中華大二)這兩個群組的受測者在作答題組最後一題時，所花費的時間都比題組內第一題的時間來得長，可以解釋當資訊檢索越多，使用者必須花費越多的時間來處理這些資訊，但 JOIN 題組(圖 4-7)在此觀點的表現並不明顯。

群組 B 在這四個題組的平均時間線型與群組 A、B 所表現的時間線型完全不同，原因可能與受測者作答習慣與實驗題目設計有關，在圖 4-5 與圖 4-6 可以很明顯的觀察到群組 B 的受測者在題組內第一題花費較多的時間，由實驗過程中的觀察得知群組 B 的受測者在作答前會較仔細的去查看整個作答的環境，包括題目的實體關係模型與關連式資料模型，而研究題目設計的方式為題組內第一題運算元最少之後的題目陸續疊加，導致題組內最後一題的答案會是前面題目答案的綜合，因此當環境熟悉後，會加快作答速度，此一原因也是導致 JOIN 題組時間曲線不符的原因。

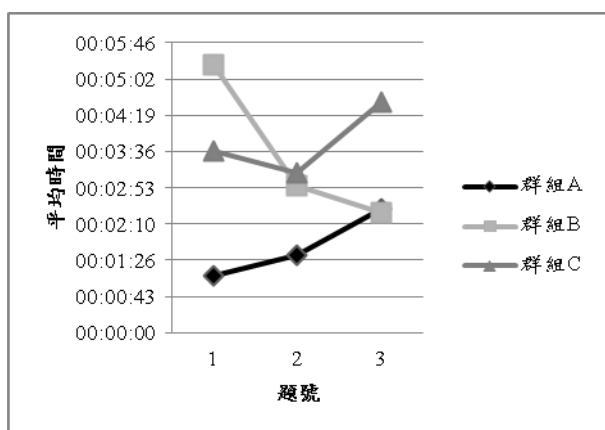


圖 4-5 SELECT 題組平均作答時間

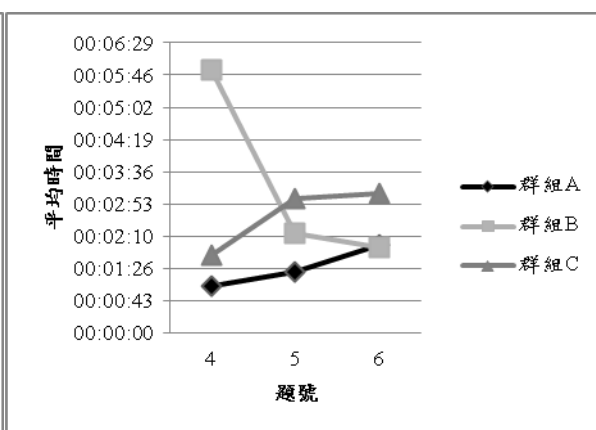


圖 4-6 WHERE 題組平均時間

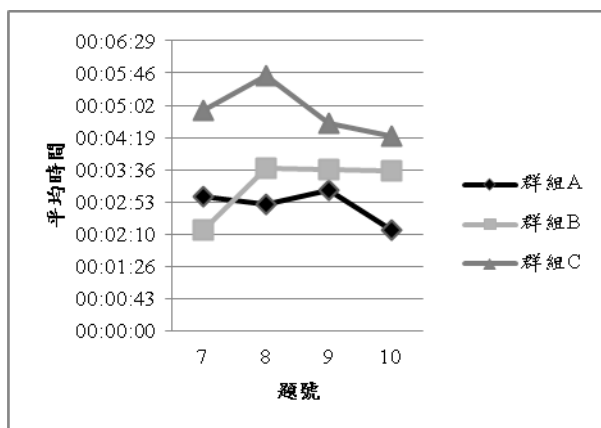


圖 4-8 SUB QUERY 題組平均作答時間

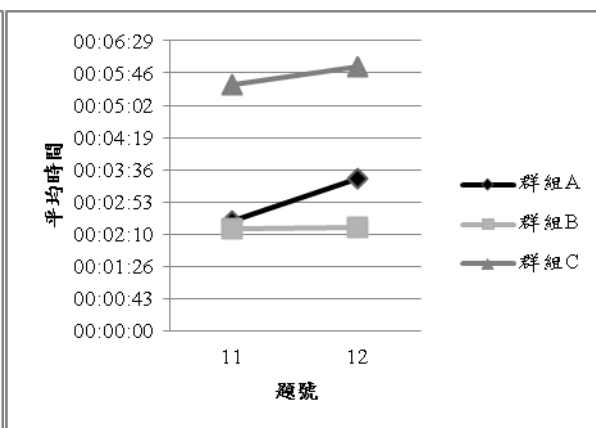


圖 4-7 JOIN 題組平均作答時間

4-1-3 題組內平均信心度分析

受測者在作答結束後，測驗系統會要求受測者填寫對此題答案的信心度，最高為 5，最低為 1，分數越高，代表受測者對於本題答案正確的信心越強。本研究完成各群組對於每題的平均信心度後，將三個不同群組的受測者資料整理如下圖 4-9、圖 4-10、圖 4-11 及圖 4-12 後做比較。

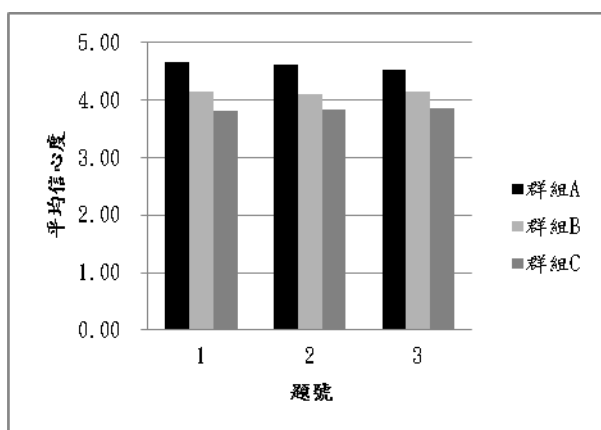


圖 4-9 SELECT 題組平均信心度

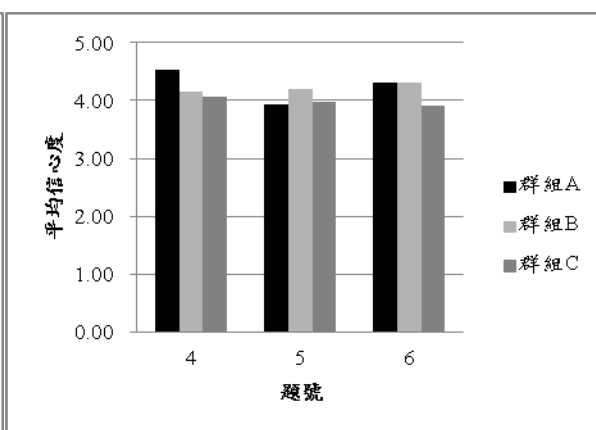


圖 4-10 WHERE 題組平均信心度

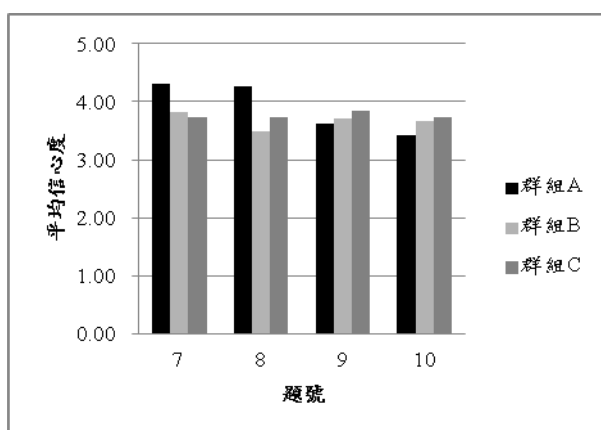


圖 4-11 JOIN 題組平均信心度

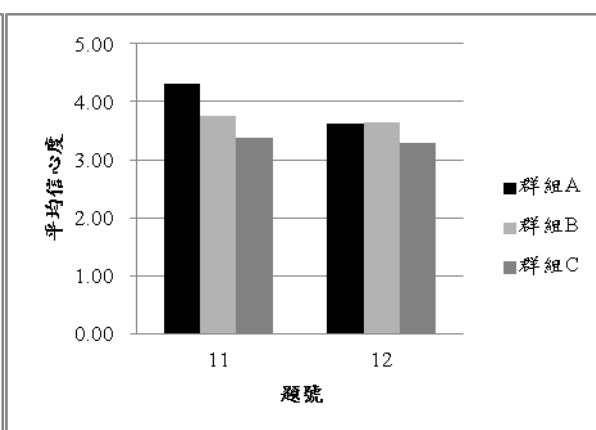


圖 4-12 SUB QUERY 題組平均信心度

信心度方面根據研究題目設計，題組中最後一題，信心度應該要比第一題信心度低，但在各題組中表現的都不是很明顯，本研究認為應該是受到受測者在填寫信心度時較為保守，以及實驗題目設計的關係，因題組中答案太過相似，因此會產生像圖 4-9 受

測者對於題組內每一題的信心度皆相似的情況。

4-1-4 題組內各衡量方法數值分析

根據各題組平均分數分析的結果，本研究的實驗題目能夠有效衡量撰寫 SQL 查詢敘述的任務複雜度。根據學理，查詢績效應隨著運算元的增加而下降，代表任務複雜度上升，所以衡量方法計算出的數值，應隨著題組內運算元的增加而增加。圖 4-13、圖 4-14 及圖 4-16 可以看到運算子權重法在題組內所算的數值皆相同，無法明顯的表現出運算元在 SQL 查詢任務中對任務複雜度的影響；Halstead 及 Enhanced Halstead 方法則符合題組平均分數圖的走向，隨著運算元的增加所計算的數值皆會上升。

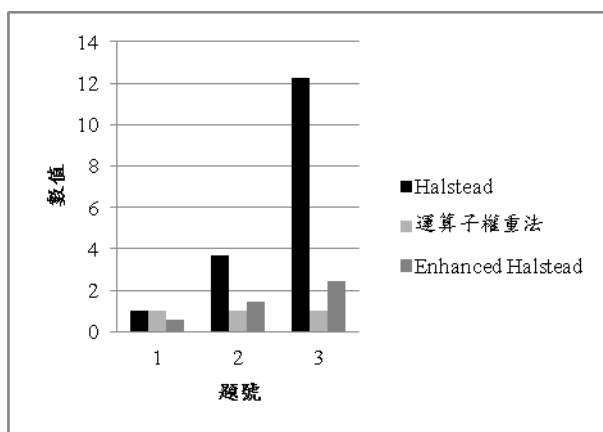


圖 4-13 SELECT 題組各衡量方法數值

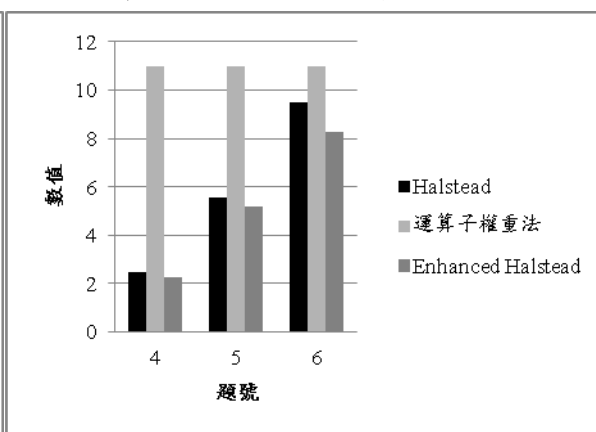


圖 4-14 WHERE 題組各衡量方法數值

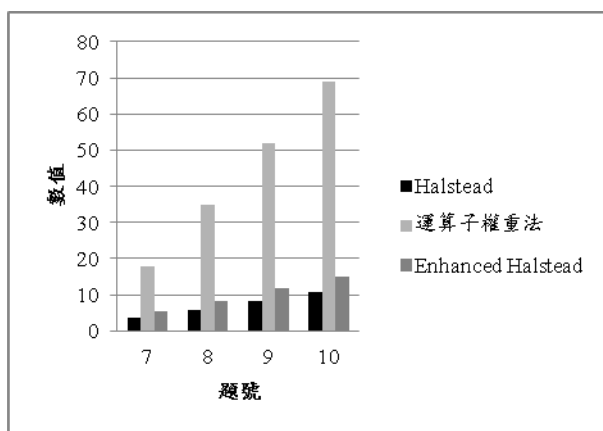


圖 4-15 JOIN 題組各衡量方法數值

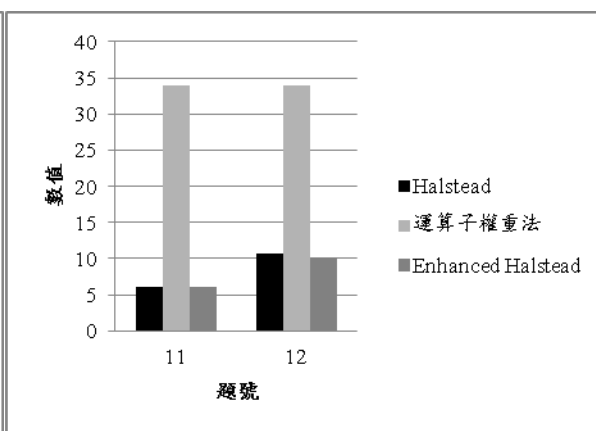


圖 4-16 SUB QUERY 題組各衡量方法數值

4-2 題組間資料分析

本節所進行的題組間資料分析將同樣數量運算元的題目群組為同一分析群組，主要目的是要了解運算子對於 SQL 查詢敘述複雜度是否具有影響力。因此資料分析將依照題目運算元的數量進行，第一組為第 1、4 及 11 題同為使用 1 個運算元的題目，第二組為第 7 及 12 題同為使用 2 個運算元的題目，第三組為第 5 及 8 題同為使用 3 個運算元的題目，最後為第 2、6 及 10 題為一組，同為使用 5 個運算元的題目，以下將按照上述分析分組順序進行分析。

將每一個分析分組所有衡量方法所獲得任務複雜度衡量數值繪如圖 4-17 至圖

4-20。在這些圖中可以觀察到三個衡量方法在同數量運算元時均表現出數值的差別，特別是運算子權重法因為只考慮運算子的權重，所以在這樣的題目分組下數值差距非常的明顯。

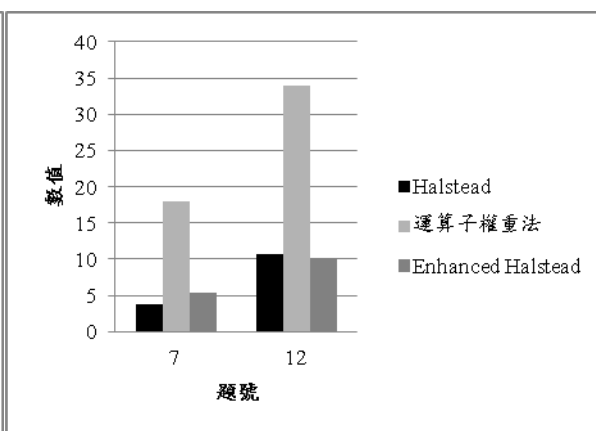
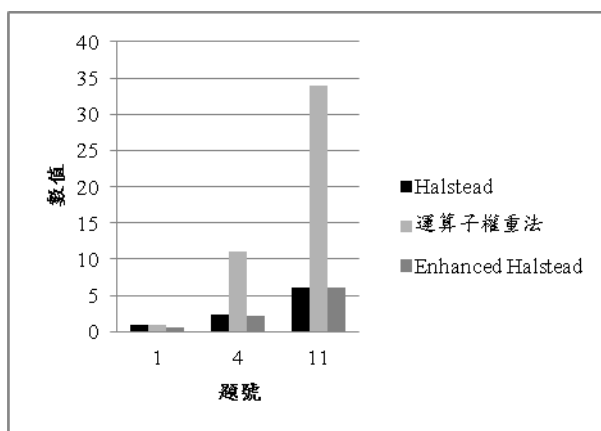


圖 4-17 1 運算元題目各衡量方法數值長條圖 圖 4-18 2 運算元題目各衡量方法數值長條圖

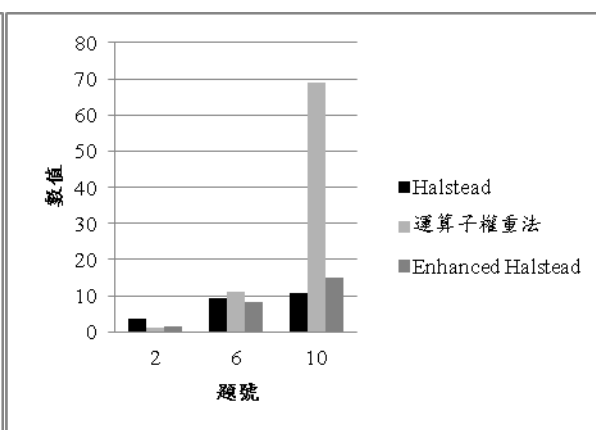
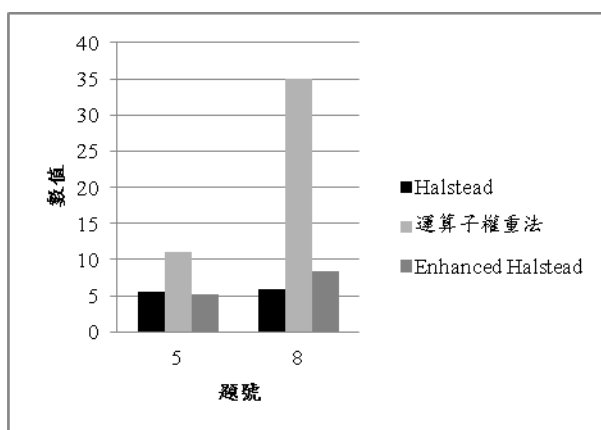


圖 4-19 3 運算元題目各衡量方法數值長條圖 圖 4-20 5 運算元題目各衡量方法數值長條圖

4-3 順序維度資料分析

本節將針對各衡量方法與受測者實際的任務複雜度感受來作比較，依序針對時間基準序列、信心度基準序列、正確性基準序列分別與運算子權重序列、Halstead 序列、Enhanced Halstead 序列做順序維度的分析與比較，以求找出可以最有效衡量出 SQL 查詢敘述任務複雜度的方法。

本節所使用之衡量方法序列分別為：

運算子權重法：1、2、3、4、5、6、7、11、12、8、9、10

Halstead 方法：1、4、7、2、5、8、11、9、6、12、10、3

Enhanced Halstead 方法：1、2、4、3、5、7、11、6、8、12、9、10

4-3-1 時間基準序列順序資料分析

表 4-1 為受測者群組每題的平均作答時間與三種衡量方法所算出的任務複雜度數值。本研究利用各群組平均作答時間由大到小排序為時間基準序列，各群組時間基準序列如下所示：

群組 A：4、1、5、2、6、10、11、3、8、7、9、12

群組 B：6、5、7、11、12、3、2、10、9、8、1、4

群組 C：4、5、6、2、1、10、3、9、7、11、8、12

將此排序作為檢視衡量方法是否符合受測者實際對於 SQL 查詢敘述任務複雜度感受的依據，以下將根據各受測者群組時間基準序列做個別順序維度方法的分析。

表 4-1 平均作答時間與衡量方法數值資料表

題號	平均作答時間			衡量方法數值		
	群組 A	群組 B	群組 C	運算子權重法	Halstead	Enhanced Halstead
1	00:01:07	00:05:20	00:03:36	1.00	1	0.56
2	00:01:32	00:02:55	00:03:11	1.00	3.71	1.45
3	00:02:28	00:02:23	00:04:35	1.00	12.28	2.45
4	00:01:02	00:05:53	00:01:44	11.00	2.45	2.23
5	00:01:21	00:02:14	00:03:01	11.00	5.57	5.17
6	00:01:57	00:01:54	00:03:07	11.00	9.51	8.29
7	00:02:59	00:02:14	00:04:56	18.00	3.7	5.40
8	00:02:50	00:03:38	00:05:42	35.00	5.9	8.42
9	00:03:08	00:03:36	00:04:38	52.00	8.26	11.67
10	00:02:15	00:03:34	00:04:21	69.00	10.78	15.09
11	00:02:27	00:02:16	00:05:30	34.00	6.06	6.06
12	00:03:25	00:02:18	00:05:54	34.00	10.68	10.25

從表 4-1 中列出表 4-2 為順序維度資料分析的結果，由表中可以看到各衡量方法序列與時間基準序列的排序差異程度都很大，尤其是在群組 B，已經可以說衡量方法序列與時間基準序列完全不一樣，導致這樣情況的可能原因應該與前面 4-1-2 所述的原因相同，受測者可能受熟悉作答環境之影響，導致研究所量測的時間不夠準確，因此出現時間基準序列與三種衡量方法序列排序皆不相關的情況。

表 4-2 平均時間順序維度資料分析結果表

群組 \ 衡量方法	群組 A		群組 B		群組 C	
	差異個數	R 值	差異個數	R 值	差異個數	R 值
運算子權重法	20	0.303	34	0.515	27	0.409
Halstead	19	0.288	40	0.606	26	0.394
Enhanced Halstead	25	0.379	38	0.576	21	0.318

4-3-2 信心度基準序列順序維度資料分析

表 4-3 為受測者群組每題的平均信心度與三種衡量方法所算出的任務複雜度數值，本研究利用各群組信心度由小到大排序為信心度基準序列，各群組信心度基準序列如下所示：

群組 A：1、2、3、4、6、7、11、8、5、9、12、10

群組 B：6、5、1、3、4、2、7、11、9、10、12、8

群組 C：4、5、6、3、9、2、1、8、10、7、11、12

將此序列作為檢視衡量方法是否符合受測者實際對於 SQL 查詢敘述任務複雜度感受的依據，以下將根據各受測者群組信心度基準序列做各別順序維度方法的分析。

表 4-3 平均信心度與衡量方法數值資料表

題號	平均信心度			衡量方法數值		
	群組 A	群組 B	群組 C	運算子權重法	Halstead	Enhanced Halstead
1	4.65	4.16	3.82	1.00	1	0.56
2	4.62	4.11	3.83	1.00	3.71	1.45
3	4.54	4.16	3.85	1.00	12.28	2.45
4	4.54	4.16	4.07	11.00	2.45	2.23
5	3.92	4.21	3.98	11.00	5.57	5.17
6	4.31	4.32	3.91	11.00	9.51	8.29
7	4.31	3.83	3.73	18.00	3.7	5.40
8	4.27	3.50	3.74	35.00	5.9	8.42
9	3.62	3.72	3.85	52.00	8.26	11.67
10	3.42	3.67	3.74	69.00	10.78	15.09
11	4.31	3.76	3.38	34.00	6.06	6.06
12	3.62	3.65	3.29	34.00	10.68	10.25

表 4-4 為順序維度資料分析的結果，由表中可以看到各衡量方法的序列與信心度基準序列的排序差異成度，群組 A 中運算子權重序列與 Enhanced Halstead 序列相似的程度很高，代表這兩個序列與信心度基準序列排序相似，但 Halstead 序列與信心度基準序列的差異就比較大；群組 B 中雖然運算子權重法序列與 Enhance Halstead 序列與信心度基準序列的差異變大了，但是差異程度還是以 Halstead 序列為最大；群組 C 則呈現與時間基準序列一樣的情況，三個衡量方法序列與信心度基準序列差異程度皆顯著的情況，原因可能與受測者填寫信心度太過集中無法明顯的表示出題目間差異有關。

表 4-4 平均信心度順序維度資料分析結果表

衡量方法	群組 A		群組 B		群組 C	
	差異個數	R 值	差異個數	R 值	差異個數	R 值
運算子權重法	12	0.182	18	0.273	28	0.424
Halstead	19	0.288	30	0.455	29	0.439
Enhanced Halstead	8	0.121	19	0.288	24	0.364

4-3-3 正確性基準序列順序維度資料分析

表 4-5 為三個受測者群組每題的平均分數與三種衡量方法所算出的任務複雜度數值，本研究利用各群組平均分數由大到小排序以此排序為正確信基準序列，個群組正確性基準序列如下所示：

群組 A：1、2、4、3、5、7、11、6、9、12、8、10

群組 B：2、3、1、4、5、6、7、11、8、12、10、9

群組 C：3、2、7、1、4、5、8、6、11、9、12、10

將此序列作為檢視衡量方法是否符合受測者實際對於 SQL 查詢敘述任務複雜度感受的依據，以下將根據受測者群組做各別順序維度方法的分析。

表 4-5 平均分數與衡量方法數值資料表

題號	平均分數			衡量方法數值		
	群組 A	群組 B	群組 C	運算子權重法	Halstead	Enhanced Halstead
1	10.00	8.95	9.27	1.00	1	0.56
2	9.90	9.47	9.40	1.00	3.71	1.45
3	9.71	9.34	9.68	1.00	12.28	2.45
4	9.90	8.82	9.18	11.00	2.45	2.23
5	9.13	8.55	7.68	11.00	5.57	5.17
6	8.46	7.76	6.59	11.00	9.51	8.29
7	8.94	7.50	9.36	18.00	3.7	5.40
8	7.69	6.53	7.45	35.00	5.9	8.42
9	7.98	5.69	6.11	52.00	8.26	11.67
10	6.44	6.11	4.10	69.00	10.78	15.09
11	8.75	6.91	6.30	34.00	6.06	6.06
12	7.79	6.32	4.64	34.00	10.68	10.25

表 4-6 為順序維度資料分析的結果，由表中可以看到運算子權重法序列與 Enhanced Halstead 序列，與各群組正確性基準序列的差距都比 Halstead 方法的差距來得小，尤其以群組 A 與群組 B 兩個群組差距最大，運算子權重法序列與 Enhanced Halstead 序列幾乎與群組 A 及群組 B 的正確性基準序列一樣，群組 C 雖然此兩個方法序列與群組 C 正確性基準序列不像群組 A 及群組 B 中這麼相似，但所衡量出的數值序列，還是比 Halstead 序列來得接近受測者的正確性基準序列。

表 4-6 平均分數順序維度資料分析結果表

群組 衡量方法	群組 A		群組 B		群組 C	
	差異個數	R 值	差異個數	R 值	差異個數	R 值
運算子權重法	12	0.182	9	0.136	15	0.227
Halstead	17	0.258	22	0.333	18	0.273
Enhance Halstead	3	0.045	6	0.091	12	0.182

4-4 資料分析結果

前三小節分別做了題組內資料分析、題組間資料分析與順序維度資料分析，三種資料分析，而每一種資料分析各別做了三個受測者群組的答題時間、答題信心度及答題正確性的分析，因為這三個因素是影響使用者 SQL 查詢敘述任務績效的因素，而使用者

SQL 使用者任務績效直接關係到使用者 SQL 查詢敘述任務複雜度，因此本研究以此方式做分析。

題組內資料分析可以看到 Halstead 方法與 Enhanced Halstead 方法在衡量這類型題目複雜度時，能夠明顯的表示出運算元數量不同時 SQL 查詢敘述任務複雜度的變化，但運算子權重法在這類型的題目中則無法表現出差別。

題組間資料分析觀察在此題組分析中皆可以表現出任務複雜度的差別，但其中以運算子權重法，因為使用的量表數值過大的關係，在衡量某些題目時，會產生數值快速上升的情況，與另兩種衡量方法形成強烈的對比。

順序維度資料分析目的在於觀察各衡量方法序列與受測者群組的三種基準序列是否相似，藉此來觀察哪一種衡量方法的數值能夠最符合 SQL 查詢敘述使用者實際對於 SQL 查詢敘述任務複雜度的感受，表 4-7 為順序維度資料分析總表，由表中可以看到衡量方法對與各群組基準序列做順序維度分析的結果，可以發現運算子權重法及 Enhanced Halstead 方法比 Halstead 方法更貼近受測者實際任務複雜度的感受，但運算子權重法在題組內分析及題組間分析皆有指出缺點，綜合以上分析結果，能最有效衡量受測者實際任務複雜度感受的衡量方法為結合運算子權重與運算元數量的 Enhanced Halstead 方法。

表 4-7 順序維度資料分析總表

直 群組	衡量方法	時間基準序列			信心度基準序列			正確性基準序列		
		運算子加權法	Halstead	Enhanced Halstead	運算子加權法	Halstead	Enhanced Halstead	運算子加權法	Halstead	Enhanced Halstead
群組 A		0.303	0.288	0.379	0.182	0.288	0.121	0.182	0.258	0.045
群組 B		0.515	0.606	0.576	0.273	0.455	0.288	0.136	0.333	0.091
群組 C		0.409	0.394	0.318	0.424	0.439	0.364	0.227	0.273	0.182

5. 結論與建議

本研究所提出的 Enhanced Halstead 方法採用同時考慮運算子與運算元的形式，並加上運算子的權重後計算 SQL 查詢敘述任務複雜度，Halstead 方法也是同時考慮運算子與運算元的形式，但並無運算子權重的概念，而運算子權重法則是完全只經由運算子的權重來決定 SQL 查詢敘述任務複雜度，本研究比較三個方法優劣的方式為將三個方法所算出的數值由小到大依序排列後得到三個方法各自的序列，將這三個序列與受測者三個群組各自的基準序列做順序維度資料分析，經由資料分析的結果證明，本研究所提之 Enhanced Halstead 衡量方法順序維度普遍上比其他兩個衡量方法來得小，代表 Enhanced Halstead 序列，與受測者基準序列的相異程度較小，衡量出的 SQL 查詢敘述任務複雜度，較能代表使用者實際任務複雜度的感受。

另一方面有關於題目設計，未來研究可以參考本研究之題目設計方法，因此方法能夠有效幫助研究者分析衡量方法是否能夠有效的衡量 SQL 查詢敘述任務複雜度中運算子與運算元所造成的差距，但建議可以提高題組間相對的難度，以避免如本研究中資料

各題組間數據差距程度過低的情況，適當的提升題組間難度，將能夠更有效的表現出題組間資料的差距。

本研究所提出之 Enhanced Halstead 方法雖然能夠有效的衡量使用者撰寫 SQL 查詢敘述任務複雜度，但此方法必須依賴運算子權重量表的數值計算 SQL 查詢敘述任務複雜度，因此未來可繼續研究衡量 SQL 查詢敘述任務複雜度的方法，發展一個不需要查詢運算子權重表而又可以同時考慮運算子權重與運算元數量的衡量公式。

參考文獻

1. 袁輝偉(2010)，以運算子權重法衡量結構化查詢語言任務複雜度，中華大學，資訊管理學系，Working Paper。
2. Borthick, A. F, Bowen, P. L., Jones, D. R., and Tse, M. H. K. (2001), "The Effects of Information Request Ambiguity and Construct Incongruence on Query Development," *Decision Support System*, Vol. 32, 3–25.
3. Bowen, P. L., O'Farrell, R.A., and Rohde, F. H. (2009), "An Empirical Investigation of End-User Query Development: The Effects of Improved Model Expressiveness vs. Complexity," *Information Systems Research*, Vol. 20, No. 4, 565–584.
4. Campbell D.J. (1988), "Task Complexity: A Review and Analysis," *Academy of Management Review*, Vol. 13, No. 1, 40 – 52.
5. Chan, H. C., Wei, K. K., and Siau, K. L. (1993), "User-Database Interface: The Effect of Abstraction Levels on Query Performance," *MIS Quarterly*, Vol. 17, 441–459.
6. Chan, H. C., Tan, B. C. Y., and Wei, K. K. (1999), "Three Importment Determinants of User Performance for Database Retrieval," *International Journal Human-Computer Studies*, Vol. 51, 895–918.
7. Halstead, M.H. (1977), *Elements of Software Science*. Elsevier, Amsterdam.
8. Liu, P. and Li, Z. (2011), "Toward Understanding the Relationship between Task Complexity and Task Performance," *Internationalization, Design, HCII 2011, LNCS 6775*, 192–200.
9. Maynard, D. C., and Hakel, M. D. (1997), "Effects of Objective and Subjective Task Complexity on Performance," *Human Performance*, Vol.10, 303 – 330.
10. Reisner, P. (1977), "Use of Psychological Experimentation as an Aid to Development of a Query Language," *IEEE Transactions on Software Engineering*, Vol. SE–3, No 3.
11. Reisner, P. (1981), "Human Factors Studies of Database Query Languages: A Survey and Assessment," *ACM Computing Surveys*, Vol. 13, No 3, 13–31.
12. Topi, H., Valacich, J. S., and Hoffer, J. A. (2005), "The Effects of Task Complexity and Time Availability Limitations on Human Performance in Database Query Tasks," *International Journal Human-Computer Studies*, Vol. 62, 349–379.
13. Wood, R.E. (1986), "Task Complexity: Definition of the construct," *Orgrnization Behavior and Human Decision Process*, Vol.37, 60–82.

Analyzing and Measuring the Task Complexity of Coding SQL

Statements

Chun-Wei Hsu¹ Chi-Chung Lee² Joseph Chiao-Wang Shih³ Hsun-Chang Chung⁴

¹Department of Information Management, Chung Hua University, Email: M09710008@chu.edu.tw

² Department of Information Management, Chung Hua University, Email: leecc@chu.edu.tw

³ Department of Information Management, Lunghwa University of Science and Technology, Email:
joseph@mail.lhu.edu.tw

⁴Department of Information Management, Chung Hua University, Email: M09810017@chu.edu.tw

ABSTRACT

Nowadays, businesses store their operational data in the database and therefore the SQL statement coding has been become a critical task for of database application system developers. To evaluate the performance of the developers, the issue of measuring the task complexity of coding SQL statements is necessary to be raised. However, little work has been done on this issue. Only two methods, that is, Halstead method and operator weight method, had been proposed to measure the task complexity of coding the SQL statement. Unfortunately, the measured task complexities of coding SQL statements on the use of Halstead method and operator weight method are not consistent with that of developers actually performed. For the reason, this study proposed a method which its measured task complexities were consistent with those of the developers actually performed and called the proposed method the enhanced Halstead method.

To evaluate the performance of the proposed method, the experiments were performed to compare the effectiveness among the enhanced Halstead, the Halstead, and the operator weight methods in terms of the order degree. The results showed that the proposed method, i.e. the enhanced Halstead method, obtained the largest order degree. This conformed that the measured task complexity of coding SQL statements on the use of the enhanced Halstead method was closer to the task complexity of the developers actually performed than those on the use of the Halstead, and the operator weight methods. This study thus concludes that the enhanced Halstead method is able to measure the task complexity of coding the SQL statements effectively.

Keywords: Structured Query Language, task complexity, Halstead method, operator weight method, Enhanced Halstead method