



(21)申請案號：106104300

(22)申請日：中華民國 106 (2017) 年 02 月 09 日

(51)Int. Cl. : G06F9/44 (2006.01)

G06F17/00 (2006.01)

(30)優先權：2016/02/29 中國大陸

201610113706.8

(71)申請人：阿里巴巴集團服務有限公司(香港地區) ALIBABA GROUP SERVICES LIMITED
(HK)

香港

(72)發明人：黃光遠(CN)；陳德品(CN)；劉蘇昱(CN)

(74)代理人：林志剛

申請實體審查：無 申請專利範圍項數：18 項 圖式數：7 共 40 頁

(54)名稱

一種應用程式的分類方法和裝置

(57)摘要

本發明實施例提供了一種應用程式的分類方法和裝置，所述方法包括：計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；依據所述關聯係數，構建針對所述待分類應用程式的節點圖；對所述節點圖進行分割，獲得一個或多個分類圖，便於充分借鑒目前已知的多個分類體系的經驗，進行融合，無需人工操作，可以避免由於人工操作導致分類結果主觀性較強，或運營人員考慮不充分等風險，從而能夠產生一份較為全面的新的應用程式分類體系。

指定代表圖：

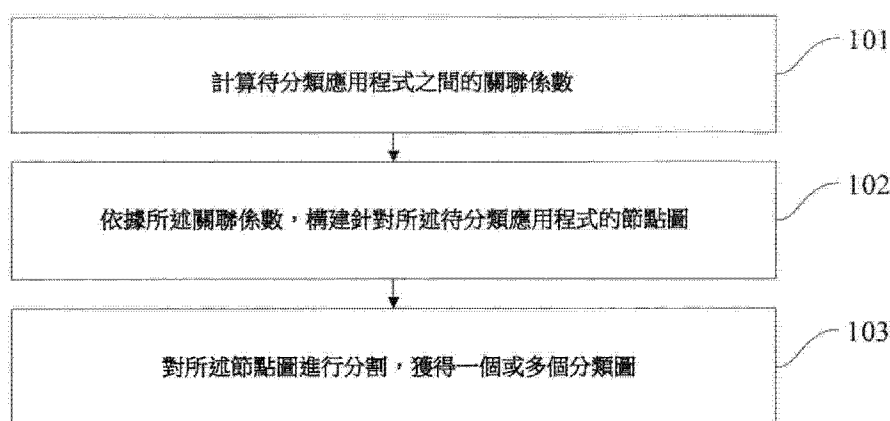


圖 1

發明摘要

※申請案號：106104300

※申請日：106年02月09日

※IPC分類：*G06F9/44* (2006.01)
G06F17/00 (2006.01)

【發明名稱】(中文/英文)

一種應用程式的分類方法和裝置

【中文】

本發明實施例提供了一種應用程式的分類方法和裝置，所述方法包括：計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；依據所述關聯係數，構建針對所述待分類應用程式的節點圖；對所述節點圖進行分割，獲得一個或多個分類圖，便於充分借鑒目前已知的多個分類體系的經驗，進行融合，無需人工操作，可以避免由於人工操作導致分類結果主觀性較強，或運營人員考慮不充分等風險，從而能夠產生一份較為全面的新的應用程式分類體系。

【英文】

【代表圖】

【本案指定代表圖】：第(1)圖。

【本代表圖之符號簡單說明】：無

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：無

發明專利說明書

(本說明書格式、順序，請勿任意更動)

【發明名稱】(中文/英文)

一種應用程式的分類方法和裝置

【技術領域】

本發明係關於計算機處理技術領域，特別是係關於一種應用程式的分類方法和一種應用程式的分類裝置。

【先前技術】

隨著智慧終端設備的普及，提供給終端設備使用的各類應用程式也是越來越多，用戶可以通過各類應用市場獲取到所需的應用程式，例如，豌豆莢，華為應用市場等。

目前，為了方便用戶下載，應用市場中的各類應用程式都是按照一定的類別進行分類的。但是，對於不同的體系，往往存在不同的分類方法，例如，豌豆莢的分類體系與華為應用市場的分類體系就不完全一致，從而可能存在應用程式的類目總數目、分類層級等均不同。因此，如果需要對各類應用程式進行管理時，單一的某個分類體系可能無法滿足我們的需求，仍然需要由運營人員進行人工歸類，使得分類操作工作量大，分類結果主觀性較強，且存在考慮不充分等風險。

【發明內容】

鑒於上述問題，提出了本發明實施例以便提供一種克服上述問題或者至少部分地解決上述問題的一種應用程式的分類方法和相應的一種應用程式的分類裝置。

為了解決上述問題，本發明公開了一種應用程式的分類方法，包括：

計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；

依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

對所述節點圖進行分割，獲得一個或多個分類圖。

可選地，所述計算待分類應用程式之間的關聯係數的步驟包括：

分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

可選地，採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortes_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

可選地，所述依據所述關聯係數，構建針對所述待分類應用程式的節點圖的步驟包括：

將所述關聯係數作為所述待分類應用程式之間邊的權重；

按照所述邊的權重，構建針對所述待分類應用程式的節點圖。

可選地，所述對所述節點圖進行分割，獲得一個或多個分類圖的步驟包括：

判斷所述節點圖中邊的權重是否大於預設閾值；

若是，則保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

可選地，所述對所述新的節點圖進行分割，獲得所述一個或多個分類圖的步驟包括：

在所述新的節點圖中，對每個待分類應用程式配置標籤；

將每個待分類應用程式的標籤傳遞至相連的待分類應用程式；

從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前是否小於預設的最大迭代次數；

若是，則返回執行所述將每個待分類應用程式的標籤傳遞至相連的待分類應用程式的步驟；

若否，則將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

可選地，所述方法還包括：

對所述一個或多個分類圖進行合併；或者，

對所述一個或多個分類圖進行再拆分。

可選地，所述對所述一個或多個分類圖進行再拆分的步驟包括：

計算所述分類圖中各應用程式間的邊的介數；

刪除最大介數值所對應的邊，得到分類子圖；

判斷所述分類子圖是否為兩個連通圖；

若否，則返回計算所述分類子圖中各應用程式間的邊的介數；

若是，則停止對分類圖進行再拆分。

可選地，採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

為了解決上述問題，本發明公開了一種應用程式的分類裝置，包括

關聯係數計算模組，用於計算待分類應用程式之間的

關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；

節點圖構建模組，用於依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

節點圖分割模組，用於對所述節點圖進行分割，獲得一個或多個分類圖。

可選地，所述關聯係數計算模組包括：

最短路徑確定子模組，用於分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

關聯係數計算子模組，用於採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

可選地，採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortest_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

可選地，所述節點圖構建模組包括：

節點圖構建子模組，用於將所述關聯係數作為所述待分類應用程式之間邊的權重，按照所述邊的權重，構建針對所述待分類應用程式的節點圖。

可選地，所述節點圖分割模組包括：

邊的權重判斷子模組，用於判斷所述節點圖中邊的權重是否大於預設閾值；

第一對應邊刪除子模組，用於在所述邊的權重大於預設閾值時，保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

節點圖分割子模組，用於對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

可選地，所述節點圖分割子模組包括：

配置單元，用於在所述新的節點圖中，對每個待分類應用程式配置標籤；

傳遞單元，用於將每個待分類應用程式的標籤傳遞至相連的待分類應用程式；

選取單元，用於從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

判斷單元，用於判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前迭代次數是否小於預設的最大迭代次數；

返回單元，用於在所述待分類應用程式所擁有的標籤發生變化，或者，當前迭代次數小於預設的最大迭代次數時，返回傳遞單元執行將每個待分類應用程式的標籤傳遞至相連的待分類應用程式的步驟；

劃分單元，用於在所述待分類應用程式所擁有的標籤未發生變化，或者，當前迭代次數大於或等於預設的最大

迭代次數時，將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

可選地，所述裝置還包括：

分類圖合併模組，用於對所述一個或多個分類圖進行合併；以及，

分類圖拆分模組，用於對所述一個或多個分類圖進行再拆分。

可選地，所述分類圖拆分模組包括：

邊的介數計算子模組，用於計算所述分類圖中各應用程式間的邊的介數；

第二對應邊刪除子模組，用於刪除最大介數值所對應的邊，得到分類子圖；

連通圖判斷子模組，用於判斷所述分類子圖是否為兩個連通圖；

返回子模組，用於在所述分類子圖不為兩個連通圖時，返回邊的介數計算子模組執行計算所述分類子圖中各應用程式間的邊的介數的步驟；

停止子模組，用於在所述分類子圖為兩個連通圖時，停止對分類圖進行再拆分。

可選地，採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中

所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

與先前技術相比，本發明實施例包括以下優點：

本發明實施例通過計算待分類應用程式之間的關聯係數，並依據所述關聯係數，構建出節點圖，然後對所述節點圖進行分割，獲得一個或多個分類圖，便於充分借鑒目前已知的多個分類體系的經驗，進行融合，無需人工操作，可以避免由於人工操作導致分類結果主觀性較強，或運營人員考慮不充分等風險，從而能夠產生一份較為全面的新的應用程式分類體系。

其次，在本發明實施例中，當獲得新的應用程式分類體系後，還可以根據實際需要對新的分類圖進行評估，執行合併或再拆分操作，以保證最終獲得的分類的準確性。

【圖式簡單說明】

圖 1 是本發明的一種應用程式的分類方法實施例一的步驟流程圖；

圖 2 是已有技術中的一種應用市場的應用程式分類示意圖；

圖 3A 是本發明的一種已知的分類體系 A 的示意圖；

圖 3B 是本發明的另一種已知的分類體系 B 的示意圖；

圖 4 是本發明的一種應用程式的分類方法實施例二的步驟流程圖；

圖 5 是本發明的一種節點圖的示意圖；

圖 6 是本發明的一種分類圖的示意圖；

圖 7 是本發明的一種應用程式的分類裝置實施例的結構框圖。

【實施方式】

為使本發明的上述目的、特徵和優點能夠更加明顯易懂，下面結合附圖和具體實施方式對本發明作進一步詳細的說明。

參照圖 1，示出了本發明的一種應用程式的分類方法實施例一的步驟流程圖，具體可以包括如下步驟：

步驟 101，計算待分類應用程式之間的關聯係數；

在本發明實施例中，當需要構建一套新的應用程式分類體系時，可以基於目前已知的多個分類體系，充分借鑒它們的經驗，進行融合，從而產生一份較為全面的新的應用程式分類體系，以減少人工操作的工作量。因此，所述待分類應用程式可以是位於一個或多個已知的分類體系中的應用程式。

具體地，可以首先計算出待分類應用程式之間的關聯係數。關聯係數體現了任意兩個應用程式之間的相關性，一般來說，關聯係數越大，說明兩個應用程式之間的關聯關係越強。

在本發明的一種較佳實施例中，所述計算待分類應用程式之間的關聯係數的步驟具體可以包括如下子步驟：

子步驟 1011，分別確定任意兩個待分類應用程式在

所述一個或多個已知的分類體系中的最短路徑；

如圖 2 所示，是已有技術中的一種應用市場的應用程式分類示意圖，通常，應用程式可以分為影音圖像、系統工具、通訊社交等大類，而在各大類下，又可進一步細分為不同的小類，例如，在網上購物這一大類下，又可以分為商城、團購、導購、海淘等不同的小類，在各個小類下，則可以包括具體的應用程式，如手機淘寶、返利網、蘇甯易購等等。為了便於理解，在本發明的實施例中，採用節點圖的方式來表示應用市場中的分類情況，節點圖中的節點分別表示具體的類目或應用程式。

因此，若將各個應用程式看作是當前分類體系中的節點，則任意兩個待分類應用程式在一個分類體系中的最短路徑可以表明兩個節點之間的距離遠近，一般來說，節點之間的距離越短，則相關性越高。

如圖 3A 和圖 3B 所示，是兩種不同的分類體系 A 和 B 的示意圖，其中 e、f、h、l、m、n 分別表示不同的類目，a、b、c、d 為待分類的應用程式。以應用程式 a、b 為例，在分類體系 A 中，它們之間的最短路徑 $\text{Shortest_path}(a,b)$ 為 2，在分類體系 B 中，它們之間的最短路徑 $\text{Shortest_path}(a,b)$ 也為 2；而對於應用程式 a、c，在分類體系 A 中，應用程式 a、c 之間的最短路徑 $\text{Shortest_path}(a,c)$ 為 4，而在分類體系 B 中，它們之間的最短路徑 $\text{Shortest_path}(a,c)$ 則為 2。

在具體實現中，各個分類體系中不同節點間的最短路

徑可以通過迪傑斯特拉算法（Dijkstra 算法）獲得。

子步驟 1012，採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

在具體實現中，可以採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortest_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

因此，對於分類體系 A 和分類體系 B 中應用程式 a 與 b ，採用上述公式，可以得到 a 、 b 之間的關聯係數 $w(a,b)=1$ ；而對於應用程式 a 與 c ，則關聯係數 $w(a,c)=0.75$ 。

步驟 102，依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

在本發明實施例中，當分別獲得全部待分類應用程式之間的關聯係數後，可以依據所述關聯係數，構建出針對所述待分類應用程式的節點圖。在節點圖中，各個節點代表不同的應用程式，而節點之間的連線則可以表示應用程式間的關係。

在具體實現中，可以將根據步驟 101 獲得的各個應用程式之間的關聯係數作為所述待分類應用程式之間邊的權

重，從而按照所述邊的權重，構建出針對所述待分類應用程式的節點圖。一般來說，在上述構建出的節點圖中，節點間的權重越大，說明它們在多個分類體系中表現出來的相關性越強，例如，若在多個分類體系中，兩個應用程式一直屬同一個葉子類目中，則這兩個應用程式之間的權重可取得最大值 1。

步驟 103，對所述節點圖進行分割，獲得一個或多個分類圖。

在本發明的一種較佳實施例中，所述對所述節點圖進行分割，獲得一個或多個分類圖的步驟具體可以包括如下子步驟：

子步驟 1031，判斷所述節點圖中邊的權重是否大於預設閾值；

子步驟 1032，若是，則保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

子步驟 1033，對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

在具體實現中，可以根據實際的分類需要，首先設定一個權重閾值，然後判斷所述節點圖中邊的權重是否大於預設閾值，若是，則可以保留對應的邊，若否，則刪除對應的邊，從而獲得新的節點圖。例如，若預設閾值為 0.5，則可以首先刪除掉節點圖中凡是權重值小於 0.5 的邊，得到新的節點圖，然後再對新的節點圖繼續分割，從而獲得一個或多個分類圖。

在本發明實施例中，通過計算待分類應用程式之間的關聯係數，並依據所述關聯係數，構建出節點圖，然後對所述節點圖進行分割，獲得一個或多個分類圖，便於充分借鑒目前已知的多個分類體系的經驗，進行融合，無需人工操作，可以避免由於人工操作導致分類結果主觀性較強，或運營人員考慮不充分等風險，從而能夠產生一份較為全面的新的應用程式分類體系。

參照圖 4，示出了本發明的一種應用程式的分類方法實施例二的步驟流程圖，具體可以包括如下步驟：

步驟 401，分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

在本發明實施例中，若將各個應用程式看作是當前分類體系中的節點，則任意兩個待分類應用程式在一個分類體系中的最短路徑可以表明兩個節點之間的距離遠近，一般來說，節點之間的距離越短，則相關性越高。因此，對於待分類的應用程式，可以首先分別確定出任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑。

如圖 3A 和圖 3B 所示，是兩種不同的分類體系 A 和 B 的示意圖，其中 e、f、h、l、m、n 分別表示不同的類目，a、b、c、d 為待分類的應用程式。以應用程式 a、b 為例，在分類體系 A 中，它們之間的最短路徑 $\text{Shortest_path}(a,b)$ 為 2，在分類體系 B 中，它們之間的最短路徑 $\text{Shortest_path}(a,b)$ 也為 2；而對於應用程式 a、c，

在分類體系 A 中，應用程式 a、c 之間的最短路徑 $Shortest_path(a,c)$ 為 4，而在分類體系 B 中，它們之間的最短路徑 $Shortest_path(a,c)$ 則為 2。

在具體實現中，各個分類體系中不同節點間的最短路徑可以通過迪傑斯特拉算法（Dijkstra 算法）獲得。

步驟 402，採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數；

在具體實現中，當已經分別獲得任意兩個待分類應用程式之間的最短路徑或，可以根據所述最短路徑，採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortest_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

例如，對於分類體系 A 和分類體系 B 中應用程式 a 與 b，採用上述公式，可以得到 a、b 之間的關聯係數 $w(a,b)=1$ ；而對於應用程式 a 與 c，則關聯係數 $w(a,c)=0.75$ 。

步驟 403，依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

在本發明的一種較佳實施例中，所述依據所述關聯係數，構建針對所述待分類應用程式的節點圖的步驟具體可

以包括如下子步驟：

子步驟 4031，將所述關聯係數作為所述待分類應用程式之間邊的權重；

子步驟 4032，按照所述邊的權重，構建針對所述待分類應用程式的節點圖。

在具體實現中，當分別獲得待分類應用程式兩兩之間的關聯係數後，可以將所述關聯係數作為構建節點圖的邊的權重，即，將任意兩個應用程式進行連接作為邊，從而構建出節點圖，以應用程式之間的關聯係數作為對應邊的權重值。一般來說，在上述構建出的節點圖中，節點間的權重越大，說明它們在多個分類體系中表現出來的相關性越強，例如，若在多個分類體系中，兩個應用程式一直屬同一個葉子類目中，則這兩個應用程式之間的權重可取得最大值 1。

步驟 404，判斷所述節點圖中邊的權重是否大於預設閾值；

在具體實現中，可以根據實際的分類需要，首先設定一個權重閾值，然後判斷所述節點圖中邊的權重是否大於預設閾值，若是，則可以保留對應的邊，若否，則可以執行步驟 405，刪除對應的邊，從而獲得新的節點圖。例如，若預設閾值為 0.5，則可以首先刪除掉節點圖中凡是權重值小於 0.5 的邊，得到新的節點圖。

步驟 405，刪除對應的邊，以獲得新的節點圖；

步驟 406，對所述新的節點圖進行分割，獲得所述一

個或多個分類圖；

在具體實現中，對所述新的節點圖進行分割，可以採用社區劃分算法 FastUnfolding 的方式進行。

在本發明的一種較佳實施例中，所述對所述新的節點圖進行分割，獲得所述一個或多個分類圖的步驟具體可以包括如下子步驟：

子步驟 4061，在所述新的節點圖中，對每個待分類應用程式配置標籤；

在具體實現中，為方便計算，該標籤可以為其用戶 ID，當然，也可以採用其他方式配置標籤，如隨機配置，只要保持標籤的唯一性即可，本發明實施例對此不加以限制。

子步驟 4062，將每個待分類應用程式的標籤傳遞至相連的待分類應用程式；

子步驟 4063，從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

子步驟 4064，判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前是否小於預設的最大迭代次數；

子步驟 4065，若是，則返回執行所述將每個待分類應用程式的標籤傳遞至相連的待分類應用程式的步驟；

子步驟 4066，若否，則將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

在首次迭代中，可以隨機選擇標籤，由於核心的節點

連著其他很多外圍節點，其標籤被隨機到的機率較大，在後續的迭代過程中，核心的節點的標籤數量會增加，逐步達到穩定。

當標籤穩定或到達最大迭代次數時，具有同樣標籤的節點屬同一個用戶群體，節點的標籤即可作為該用程式的識別標籤。

例如，如圖 5 所示，是本發明的一種節點圖的示意圖，以節點的名稱作為節點的標籤，即節點 R、S、T、U 的標籤的標籤分別為 R、S、T、U，則其在迭代的過程如下：

| 迭代次數 | 節點 | 接收到的標籤 | 選擇的標籤 |
|------|----|--------|-------|
| 1 | R | S、T、U | U |
| | S | R、T | R |
| | T | R、S | S |
| | U | R | R |
| 2 | R | R、R、S | R |
| | S | S、U | S |
| | T | R、U | R |
| | U | R | R |
| 3 | R | R、R、S | R |
| | S | R、R | R |
| | T | R、S | R |
| | U | R | R |

在第 3 輪迭代後，應用程式所擁有的標籤都為 R，不再發生變化，因此，節點 R、S、T、U 對應的應用程式屬

於相同的分類，可以將其劃分在同一個分類圖中。

當然，除了上述社區劃分算法之外，還可以採用其他社區劃分算法，如 GN (Community structure in social networks) 算法、Louvain 算法等等，本發明實施例對此不加以限制。

步驟 407，對所述一個或多個分類圖進行合併；或者，對所述一個或多個分類圖進行再拆分。

在本發明實施例中，當按照上述方法獲得新的分類圖後，還可以對新的分類圖進行再處理，例如，對所述一個或多個分類圖進行合併；或者，對所述一個或多個分類圖進行再拆分。

在具體實現中，若需要將多個分類圖進行合併，可以直接合成，並可以形成層次化結構，例如，將原來的兩個分類圖 A 與 B，合成後的大分類 C 的子分類。

在本發明的一種較佳實施例中，所述對所述一個或多個分類圖進行再拆分的步驟則可以包括如下子步驟：

子步驟 4071，計算所述分類圖中各應用程式間的邊的介數；

子步驟 4072，刪除最大介數值所對應的邊，得到分類子圖；

子步驟 4073，判斷所述分類子圖是否為兩個連通圖；

子步驟 4074，若否，則返回計算所述分類子圖中各應用程式間的邊的介數；

子步驟 4075，若是，則停止對分類圖進行再拆分。

在具體實現中，可以採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

例如，如圖 6 所示，是本發明的一種分類圖的示意圖，按照上述公式，可以得到各條邊的介數值如下：

(1) AB : 1/30

(2) BC : 4/30

(3) AC : 4/30

(4) CD : 5/30

(5) DE : 4/30

(6) DF : 4/30

通過比較，可知，其中邊 CD 的介數值最大，因此刪除邊 CD，由於在刪除邊 CD 後，原分類圖被拆分成了兩個連通圖，即分類子圖 ABC 與 DEF，達到了拆分的效果，可以停止對分類圖進行再拆分。

在本發明實施例中，當獲得新的應用程式分類體系後，還可以根據實際需要對新的分類圖進行評估，執行合併或再拆分操作，以保證最終獲得的分類的準確性。

需要說明的是，對於方法實施例，為了簡單描述，故將其都表述為一系列的動作組合，但是本領域技術人員應

該知悉，本發明實施例並不受所描述的動作順序的限制，因為依據本發明實施例，某些步驟可以採用其他順序或者同時進行。其次，本領域技術人員也應該知悉，說明書中所描述的實施例均屬於較佳實施例，所涉及的動作並不一定是本發明實施例所必須的。

參照圖 7，示出了本發明的一種應用程式的分類裝置實施例的結構框圖，具體可以包括如下模組：

關聯係數計算模組 701，用於計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；

節點圖構建模組 702，用於依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

節點圖分割模組 703，用於對所述節點圖進行分割，獲得一個或多個分類圖。

在本發明實施例中，所述關聯係數計算模組 701 具體可以包括如下子模組：

最短路徑確定子模組 7011，用於分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

關聯係數計算子模組 7012，用於採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

在本發明實施例中，可以採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortes_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a, b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

在本發明實施例中，所述節點圖構建模組 702 具體可以包括如下子模組：

節點圖構建子模組 7021，用於將所述關聯係數作為所述待分類應用程式之間邊的權重，按照所述邊的權重，構建針對所述待分類應用程式的節點圖。

在本發明實施例中，所述節點圖分割模組 703 具體可以包括如下子模組：

邊的權重判斷子模組 7031，用於判斷所述節點圖中邊的權重是否大於預設閾值；

第一對應邊刪除子模組 7032，用於在所述邊的權重大於預設閾值時，保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

節點圖分割子模組 7033，用於對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

在本發明實施例中，所述節點圖分割子模組 7033 具體可以包括如下單元：

配置單元 331，用於在所述新的節點圖中，對每個待分類應用程式配置標籤；

傳遞單元 332，用於將每個待分類應用程式的標籤傳

遞至相連的待分類應用程式；

選取單元 333，用於從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

判斷單元 334，用於判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前迭代次數是否小於預設的最大迭代次數；

返回單元 335，用於在所述待分類應用程式所擁有的標籤發生變化，或者，當前迭代次數小於預設的最大迭代次數時，返回傳遞單元執行將每個待分類應用程式的標籤傳遞至相連的待分類應用程式的步驟；

劃分單元 336，用於在所述待分類應用程式所擁有的標籤未發生變化，或者，當前迭代次數大於或等於預設的最大迭代次數時，將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

在本發明實施例中，所述裝置還可以包括如下模組：

分類圖合併模組 704，用於對所述一個或多個分類圖進行合併；以及，

分類圖拆分模組 705，用於對所述一個或多個分類圖進行再拆分。

在本發明實施例中，所述分類圖拆分模組 705 具體可以包括如下子模組：

邊的介數計算子模組 7051，用於計算所述分類圖中各應用程式間的邊的介數；

第二對應邊刪除子模組 7052，用於刪除最大介數值所對應的邊，得到分類子圖；

連通圖判斷子模組 7053，用於判斷所述分類子圖是否為兩個連通圖；

返回子模組 7054，用於在所述分類子圖不為兩個連通圖時，返回邊的介數計算子模組執行計算所述分類子圖中各應用程式間的邊的介數的步驟；

停止子模組 7055，用於在所述分類子圖為兩個連通圖時，停止對分類圖進行再拆分。

在本發明實施例中，可以採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

對於裝置實施例而言，由於其與方法實施例基本相似，所以描述的比較簡單，相關之處參見方法實施例的部分說明即可。

本說明書中的各個實施例均採用遞進的方式描述，每個實施例重點說明的都是與其他實施例的不同之處，各個實施例之間相同相似的部分互相參見即可。

本領域內的技術人員應明白，本發明實施例的實施例可提供為方法、裝置、或計算機程式產品。因此，本發明實施例可採用完全硬體實施例、完全軟體實施例、或結合

軟體和硬體方面的實施例的形式。而且，本發明實施例可採用在一個或多個其中包含有計算機可用程式代碼的計算機可用儲存媒體(包括但不限於磁碟記憶體、CD-ROM、光學記憶體等)上實施的計算機程式產品的形式。

在一個典型的配置中，所述計算機設備包括一個或多個處理器(CPU)、輸入/輸出介面、網路介面和記憶體。記憶體可能包括計算機可讀媒體中的非永久性記憶體，隨機存取記憶體(RAM)和/或非易失性記憶體等形式，如唯讀記憶體(ROM)或快閃記憶體(flash RAM)。記憶體是計算機可讀媒體的示例。計算機可讀媒體包括永久性和非永久性、可行動和非可行動媒體可以由任何方法或技術來實現資訊儲存。資訊可以是計算機可讀指令、資料結構、程式的模組或其他資料。計算機的儲存媒體的例子包括，但不限於相變記憶體(PRAM)、靜態隨機存取記憶體(SRAM)、動態隨機存取記憶體(DRAM)、其他類型的隨機存取記憶體(RAM)、唯讀記憶體(ROM)、電可擦除可編程唯讀記憶體(EEPROM)、快閃記憶體或其他記憶體技術、光碟唯讀記憶體(CD-ROM)、數位多功能光碟(DVD)或其他光學儲存、磁盒式磁帶，磁帶磁碟儲存或其他磁性儲存設備或任何其他非傳輸媒體，可用於儲存可以被計算設備存取的資訊。按照本文中的界定，計算機可讀媒體不包括非持續性的電腦可讀媒體(transitory media)，如調變的資料信號和載波。

本發明實施例是參照根據本發明實施例的方法、終端

設備(系統)、和計算機程式產品的流程圖和／或方框圖來描述的。應理解可由計算機程式指令實現流程圖和／或方框圖中的每一流程和／或方框、以及流程圖和／或方框圖中的流程和／或方框的結合。可提供這些計算機程式指令到通用計算機、專用計算機、嵌入式處理機或其他可編程資料處理終端設備的處理器以產生一個機器，使得通過計算機或其他可編程資料處理終端設備的處理器執行的指令產生用於實現在流程圖一個流程或多個流程和／或方框圖一個方框或多個方框中指定的功能的裝置。

這些計算機程式指令也可儲存在能引導計算機或其他可編程資料處理終端設備以特定方式工作的計算機可讀記憶體中，使得儲存在該計算機可讀記憶體中的指令產生包括指令裝置的製造品，該指令裝置實現在流程圖一個流程或多個流程和／或方框圖一個方框或多個方框中指定的功能。

這些計算機程式指令也可裝載到計算機或其他可編程資料處理終端設備上，使得在計算機或其他可編程終端設備上執行一系列操作步驟以產生計算機實現的處理，從而在計算機或其他可編程終端設備上執行的指令提供用於實現在流程圖一個流程或多個流程和／或方框圖一個方框或多個方框中指定的功能的步驟。

儘管已描述了本發明實施例的較佳實施例，但本領域內的技術人員一旦得知了基本創造性概念，則可對這些實施例做出另外的變更和修改。所以，所附申請專利範圍意

欲解釋為包括較佳實施例以及落入本發明實施例範圍的所有變更和修改。

最後，還需要說明的是，在本文中，諸如第一和第二等之類的關係術語僅僅用來將一個實體或者操作與另一個實體或操作區分開來，而不一定要求或者暗示這些實體或操作之間存在任何這種實際的關係或者順序。而且，術語“包括”、“包含”或者其任何其他變體意在涵蓋非排他性的包含，從而使得包括一系列要素的過程、方法、物品或者終端設備不僅包括那些要素，而且還包括沒有明確列出的其他要素，或者是還包括為這種過程、方法、物品或者終端設備所固有的要素。在沒有更多限制的情況下，由語句“包括一個……”限定的要素，並不排除在包括所述要素的過程、方法、物品或者終端設備中還存在另外的相同要素。

以上對本發明所提供的一種應用程式的分類方法和一種應用程式的分類裝置，進行了詳細介紹，本文中應用了具體個例對本發明的原理及實施方式進行了闡述，以上實施例的說明只是用於幫助理解本發明的方法及其核心思想；同時，對於本領域的一般技術人員，依據本發明的思想，在具體實施方式及應用範圍上均會有改變之處，綜上所述，本說明書內容不應理解為對本發明的限制。

【符號說明】

701：關聯係數計算模組

702：節點圖構建模組

703：節點圖分割模組

申請專利範圍

1.一種應用程式的分類方法，包括：

計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分類體系中；

依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

對所述節點圖進行分割，獲得一個或多個分類圖。

2.根據請求項 1 所述的方法，其中，所述計算待分類應用程式之間的關聯係數的步驟包括：

分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

3.根據請求項 2 所述的方法，其中，採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortes_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

4.根據請求項 2 或 3 所述的方法，其中，所述依據所述關聯係數，構建針對所述待分類應用程式的節點圖的步驟包括：

將所述關聯係數作為所述待分類應用程式之間邊的權重；

按照所述邊的權重，構建針對所述待分類應用程式的節點圖。

5.根據請求項 4 所述的方法，其中，所述對所述節點圖進行分割，獲得一個或多個分類圖的步驟包括：

判斷所述節點圖中邊的權重是否大於預設閾值；

若是，則保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

6.根據請求項 5 所述的方法，其中，所述對所述新的節點圖進行分割，獲得所述一個或多個分類圖的步驟包括：

在所述新的節點圖中，對每個待分類應用程式配置標籤；

將每個待分類應用程式的標籤傳遞至相連的待分類應用程式；

從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前是否小於預設的最大迭代次數；

若是，則返回執行所述將每個待分類應用程式的標籤

傳遞至相連的待分類應用程式的步驟；

若否，則將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

7.根據請求項 1 或 2 或 3 或 5 或 6 所述的方法，其中，還包括：

對所述一個或多個分類圖進行合併；或者，

對所述一個或多個分類圖進行再拆分。

8.根據請求項 7 所述的方法，其中，所述對所述一個或多個分類圖進行再拆分的步驟包括：

計算所述分類圖中各應用程式間的邊的介數；

刪除最大介數值所對應的邊，得到分類子圖；

判斷所述分類子圖是否為兩個連通圖；

若否，則返回計算所述分類子圖中各應用程式間的邊的介數；

若是，則停止對分類圖進行再拆分。

9.根據請求項 8 所述的方法，其中，採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

10.一種應用程式的分類裝置，包括：

關聯係數計算模組，用於計算待分類應用程式之間的關聯係數，所述待分類應用程式位於一個或多個已知的分

類體系中；

節點圖構建模組，用於依據所述關聯係數，構建針對所述待分類應用程式的節點圖；

節點圖分割模組，用於對所述節點圖進行分割，獲得一個或多個分類圖。

11.根據請求項 10 所述的裝置，其中，所述關聯係數計算模組包括：

最短路徑確定子模組，用於分別確定任意兩個待分類應用程式在所述一個或多個已知的分類體系中的最短路徑；

關聯係數計算子模組，用於採用所述最短路徑，計算所述任意兩個待分類應用程式之間的關聯係數。

12.根據請求項 11 所述的裝置，其中，採用如下公式，計算所述任意兩個待分類應用程式之間的關聯係數：

$$w(a,b) = \frac{1}{n} * \sum_{n=1}^n \left(\frac{2}{shortest_path(a,b)} \right)$$

其中， $w(a, b)$ 為待分類應用程式 a 與 b 之間的關聯係數；

n 為已知的分類體系的數量；

$shortest_path(a,b)$ 為待分類應用程式 a 與 b 在已知的分類體系中的最短路徑。

13.根據請求項 11 或 12 所述的裝置，其中，所述節點圖構建模組包括：

節點圖構建子模組，用於將所述關聯係數作為所述待分類應用程式之間邊的權重，按照所述邊的權重，構建針

對所述待分類應用程式的節點圖。

14.根據請求項 13 所述的裝置，其中，所述節點圖分割模組包括：

邊的權重判斷子模組，用於判斷所述節點圖中邊的權重是否大於預設閾值；

第一對應邊刪除子模組，用於在所述邊的權重大於預設閾值時，保留對應的邊，若否，則刪除對應的邊，以獲得新的節點圖；

節點圖分割子模組，用於對所述新的節點圖進行分割，獲得所述一個或多個分類圖。

15.根據請求項 14 所述的裝置，其中，所述節點圖分割子模組包括：

配置單元，用於在所述新的節點圖中，對每個待分類應用程式配置標籤；

傳遞單元，用於將每個待分類應用程式的標籤傳遞至相連的待分類應用程式；

選取單元，用於從每個待分類應用程式接收到的標籤中，按照標籤的數量選取一個標籤作為所擁有的標籤；

判斷單元，用於判斷在所述新的節點圖中，待分類應用程式所擁有的標籤是否發生變化，或者，當前迭代次數是否小於預設的最大迭代次數；

返回單元，用於在所述待分類應用程式所擁有的標籤發生變化，或者，當前迭代次數小於預設的最大迭代次數時，返回傳遞單元執行將每個待分類應用程式的標籤傳遞

至相連的待分類應用程式的步驟；

劃分單元，用於在所述待分類應用程式所擁有的標籤未發生變化，或者，當前迭代次數大於或等於預設的最大迭代次數時，將擁有相同標籤的待分類應用程式劃分在同一個分類圖中，獲得一個或多個分類圖。

16.根據請求項 10 或 11 或 12 或 14 或 15 所述的裝置，其中，還包括：

分類圖合併模組，用於對所述一個或多個分類圖進行合併；以及，

分類圖拆分模組，用於對所述一個或多個分類圖進行再拆分。

17.根據請求項 16 所述的裝置，其中，所述分類圖拆分模組包括：

邊的介數計算子模組，用於計算所述分類圖中各應用程式間的邊的介數；

第二對應邊刪除子模組，用於刪除最大介數值所對應的邊，得到分類子圖；

連通圖判斷子模組，用於判斷所述分類子圖是否為兩個連通圖；

返回子模組，用於在所述分類子圖不為兩個連通圖時，返回邊的介數計算子模組執行計算所述分類子圖中各應用程式間的邊的介數的步驟；

停止子模組，用於在所述分類子圖為兩個連通圖時，停止對分類圖進行再拆分。

18. 根據請求項 17 所述的裝置，其中，採用如下公式，計算所述分類圖中各應用程式間的邊的介數：

$$B(e) = \frac{p}{q}$$

其中， $B(e)$ 為邊 e 對應的介數值； q 為所述分類圖中所有最短路徑的數目； p 為經過邊 e 的最短路徑的數目。

圖式

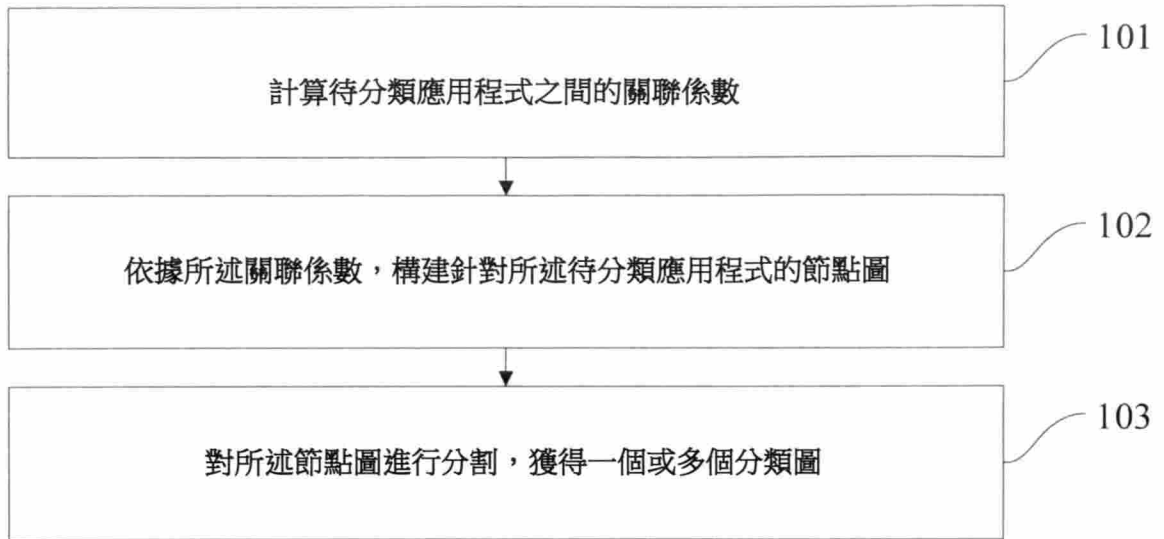


圖 1



圖 2

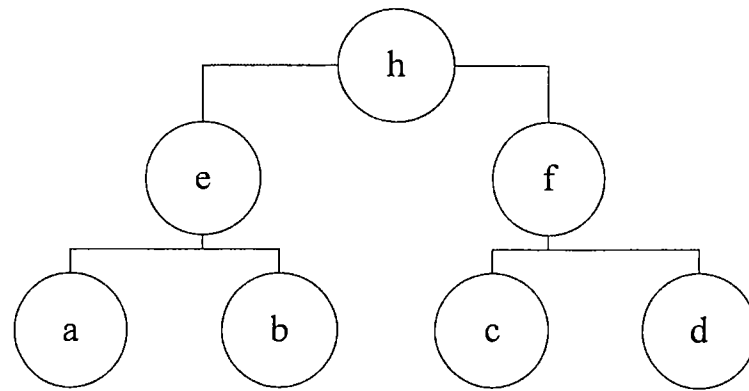


圖 3A

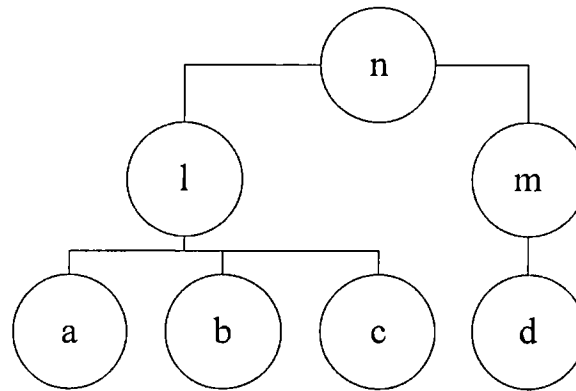


圖 3B

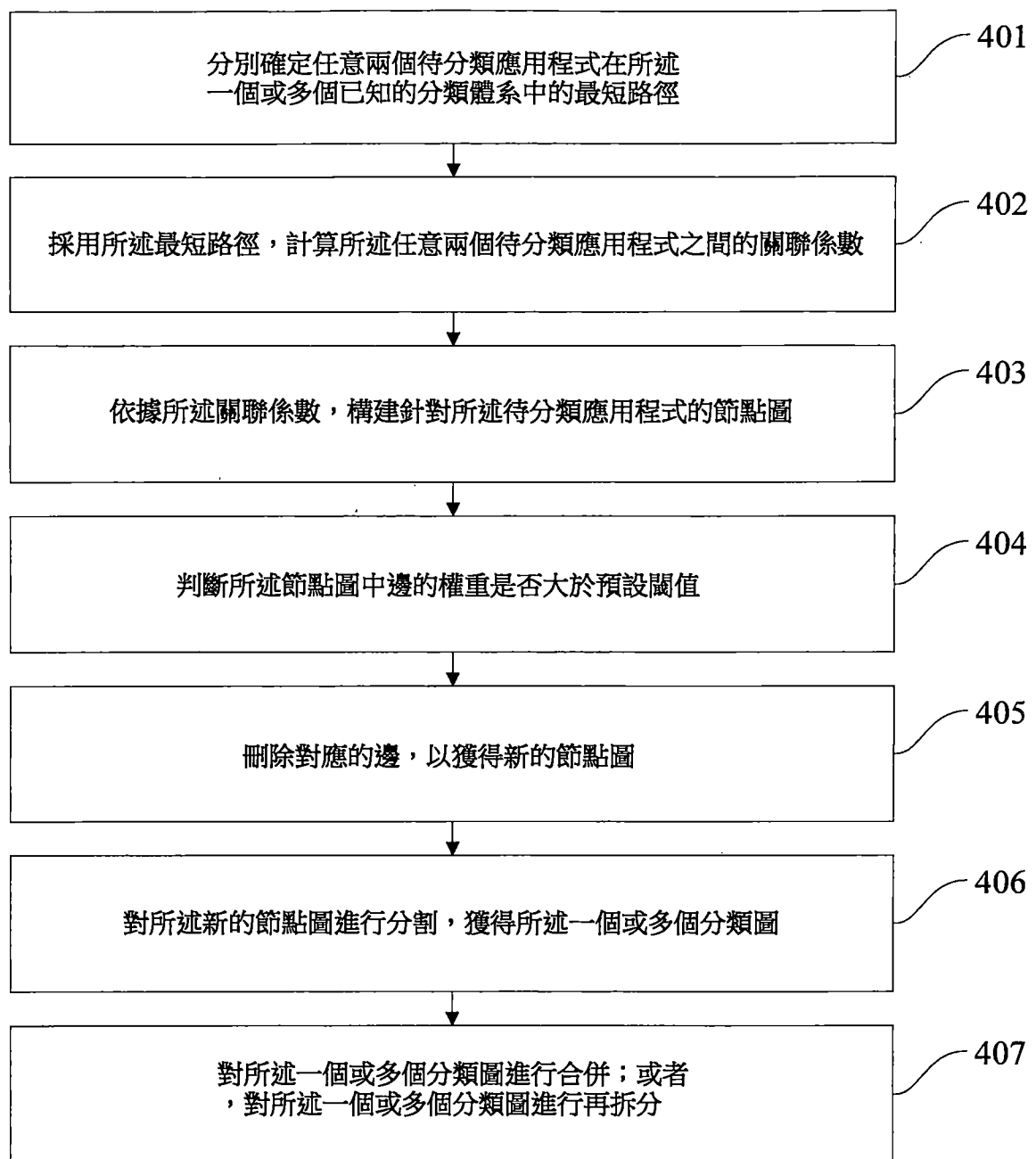


圖 4

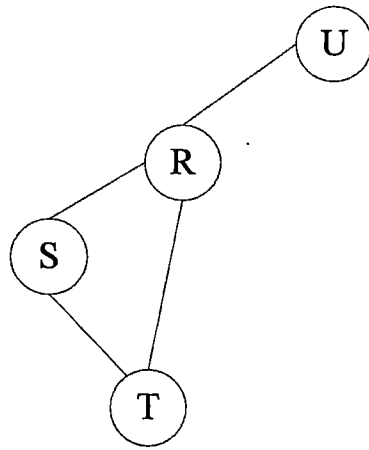


圖 5

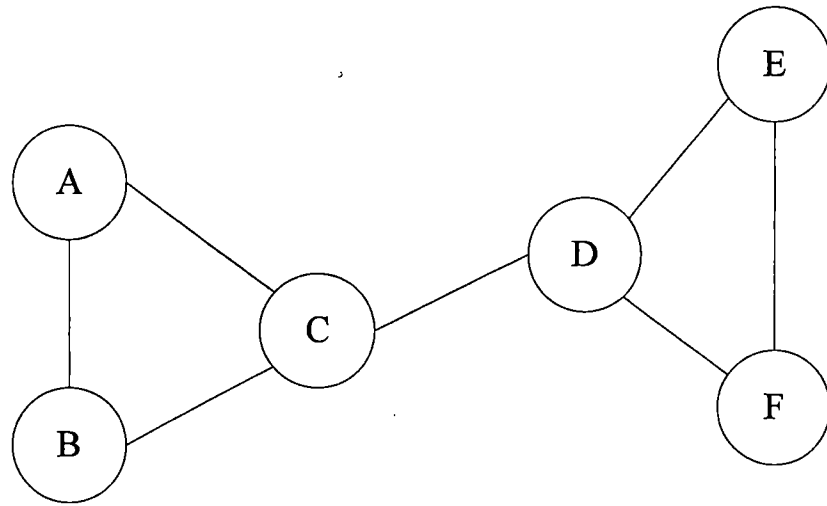


圖 6

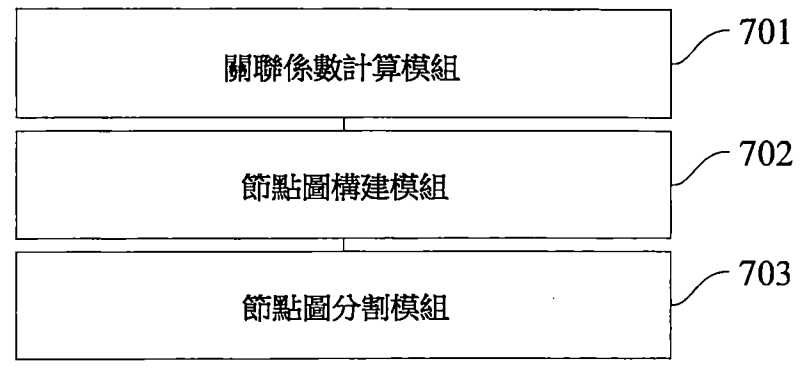


圖 7