

基於模糊推論代理人之Web 2.0府城美食知識管理平臺

李健興

國立臺南大學

資訊工程學系

leecs@mail.nutn.edu.tw

黃獻德

國立臺南大學

數位學習科技學系

tonton@dark-circles.us

莊宗嚴

國立臺南大學

數位學習科技學系

chuangyen@mail.nutn.edu.tw

摘要

國內生活型態迅速的改變，旅遊人口大幅度的增加，在旅遊中，容易忽略了飲食營養健康；另一方面，一般傳統美食介紹網站又過於呆板且制式化，並無法得知該食物的詳細營養資訊。本論文利用 Web 2.0 的社群分享概念，透過使用者的互動分享來搜集各美食的資料，並且混搭 (Mash up) Google Maps 清楚標示出所在位置，來建構府城美食知識管理平臺，另一方面應用模糊邏輯，適當的標示出使用者的身體質量指數值 (BMI) 與該食物的口感油膩度與食用該項美食後將產生的熱量，應用模糊推論代理人，因時因人而異的適當推薦給使用者，藉此增加互動，本論文將根據以上所需功能在 Ruby on Rails (ROR) 為開發語言的架構上，以 Linux, Apache2, MySQL, PHP (LAMP) 為平台實作出網路社群知識管理平臺。

關鍵詞：Web 2.0、知識管理、模糊推論、代理人、Ruby on Rails

一、前言

國際間 Web 2.0 的風潮正不斷的擴張，特別是當 Google 以近 16.5 億美金的天價購買 YouTube 後，各大媒體皆爭相報導，在 Google 以 Web 2.0 為關鍵字來搜尋也會出現約六億九千萬筆相關搜尋結果，各大企業對這股趨勢皆相當重視；因為，Web 2.0 已經開啟了一個嶄新的商業模式，進而影響產業鏈之結構，甚至創造新型態服務模式[2]。另一方面，隨著時代的進步，生活型態的改變，民眾利用假日出外旅遊大幅度增加，在旅遊行程規劃中，往往需要美食小吃的景點路線以及評論介紹資料，做行前規劃時勢必需要搜集相關資料，但是一般網站或是書籍太過於制式化的介紹旅遊景點以及美食資訊，忽略了因為生活水準的提昇，在追尋著美味的同時，營養飲食健康更是民眾注重且不可缺少卻不易獲得的資料。

根據以上所提，本論文針對民眾需求且傳統網站常常忽略的營養飲食健康；即使民眾有營養師建議的菜單，但通常都太過於枯燥，忽略了使用者對該項食物的喜好程度以及食物的口感油膩還有當時的飽足感等問題，而旅遊途中也不太可能精密計算食物熱量。而另一方面，一般傳統美食介紹網站又過於呆板且制式化，並無法得知該食物的詳細營養資訊。本論文利用 Web 2.0 的社群分享概念，

提出一個模糊推論代理人，透過使用者的互動分享來搜集各美食的資料，並且 Mash up Google Maps 的服務，可以方便清楚的標示出美食店家的所在位置，另一方面根據使用者的身材條件計算出建議的 BMI 值，再根據使用者對該項食物的喜好程度、食物的口感油膩還有當時的飽足感來進行推論並計算出適當的推薦給使用者。本論文將根據以上功能以 ROR 為開發語言，以 LAMP 為平台實作出一網路社群府城美食知識管理平臺 U-DO-EAT@Tainan 2.0。

本論文之架構如下，第二節針對系統運用之概念作相關文獻探討，第三節說明整個 U-DO-EAT@Tainan 2.0 系統的架構，第四節則簡要說明網站平臺目前實作成果，第五節為本論文的結論以及未來發展。

二、相關研究與文獻探討

(一) Web 2.0

二十一世紀，網際網路已經從以網站為中心演變成以使用者為中心，從單向傳輸演變為雙向溝通，從獨斷的內容創造變成集體的創作分享[13]。網際網路漸漸的改變了我們的生活習慣，現在回到家第一件事，很可能是上網，看看有沒有新 E-Mail，部落格上有沒有新文章等等。其實在網路世界中，網路應用一直都是持續的在演進，而 Web 2.0 所強調的是一種應用，而不是傳統的顛覆或者是取代，這就是 Web 2.0 的最大價值[10]。結合 Craigslist (如圖 1，craigslist.com，提供美國等大城市中的分類廣告列表) 與 Google Maps (如圖 2，maps.google.com) 兩大服務來提供線上房屋仲介地圖服務之功能的 HousingMaps (如圖 3，HousingMaps.com)，為目前 Web 2.0 商業行為最佳代表網站之一，O'Reilly 媒體公司總裁兼 CEO Tim O'Reilly 更把 HousingMaps 評為網路上第一個真實的 Web 2.0 應用與 Mash up 的代表作[2]。

在 HousingMaps 這樣創新的 Web 2.0 網站應用服務尚未出現之前，使用者必須先去查 Craigslist 的分類廣告，再根據查到的每筆地址，以人工方式，在地圖網站一一查詢地點和交通路線。這個情形就如同沒有做好資訊整合的企業一樣，被迫以人工、低效能的方式，在不同的系統操作介面間，做 E-mail，甚至資料剪貼、傳真等工作，相同的一筆資料得反覆重新輸入多次，而導致辦事效能低落[2]；因此透過 Web 2.0 的 Mash up 以及社群分享的概念，不僅可大幅改進其傳統的網站功能，更可

以為使用者省下許多時間，以獲得更多完整的資訊。



圖 1 sfbay.craigslist.org[6]

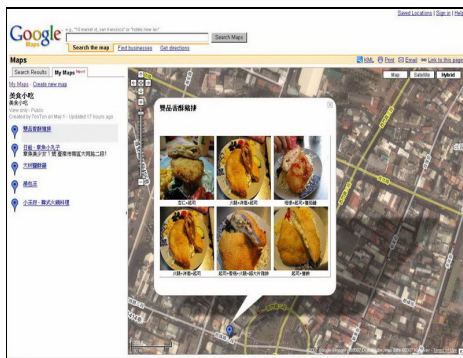


圖 2 maps.google.com[7]

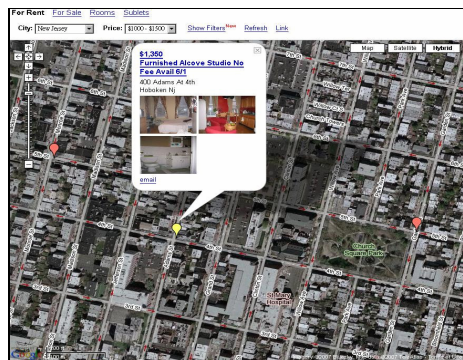


圖 3 housingmaps.com[9]

再來我們針對 Web 2.0 網站的重要精神 U.S.E.R (Unconstraint, Service, Externality, Reward) [2]簡要描述如下：

- (1) U 代表的是 Unconstraint：使用者擁有控制權且不再被限制，從原本接受訊息轉變為核心，在所有使用者的智慧與創意的貢獻下，創造了所謂的「集思廣益」。
- (2) S 代表的是 Service：相較於傳統網站，Web 2.0 網站所重視的是所提供的服務，是一個重視使用者體驗與感受而且是以服務使用者為導

向的平台。

- (3) E 代表的是 Externality：在經濟學領域中最常討論的外部性，經由眾人的參與，方便性日漸增加，隨著加入的使用者越來越多，其方便性也越來越高，不因為電腦設備的不同而有不相容的情形發生，藉由日漸增加的使用者，帶來越多的方便性與熱潮。
- (4) R 代表的是 Reward：在 Web 2.0 網站裡，只要方法對了，將可帶來驚人的獲利，Wired 雜誌的總編輯 Chris Anderson 先生提到「當可供選擇的產品極大豐富，用戶需求的多樣性及消費隨意性就格外明顯」；由於在網路上販售商品，可以提供給消費者選擇的商品種類數量，並沒有實體商店受限於成本與設備，只能參考本益比提供少數受歡迎的商品數量，整體的收益，在眾多商品數量的銷售下，反而促成驚人的成長。

關於 Web 2.0，還有幾個重要技術，它們分別是 Asynchronous JavaScript and XML (AJAX)、Rich Site Summary (RSS)、Mash up、標籤 (Tag)。以下分別進行描述：

- (1) AJAX：將客戶端與伺服器端連結起來，能夠在不更新整個頁面的前提下提供新資訊，使得網站應用服務可以快速回應，並避免在網路上發送不必要的程式碼，使得瀏覽器與伺服器之間的傳輸大為減少，加快網頁的處理速度以及顯示時間[12]。
- (2) RSS：用於共享新聞和其它 Web 內容的資料交換規範，透過 RSS 資料獨立，不再依附著網頁或者是網站上，RSS 可以比擬為訂報機，它不僅鏈結到一個網頁，而且訂閱了這個網頁，當該頁面產生變化時訂閱者將會得到通知。RSS 可用來通知新文章出現，並且包括各種更新的資料[2]。
- (3) 混搭 (Mash up)：應用在 Web 2.0 上，意指透過各種網頁服務所衍生而發展出的新網路應用服務。透過各種不同的網路應用服務及搭配不同的內容或資訊，創造出一種新的服務應用[2]。
- (4) Tag：類似 keyword，但 keyword 是內容的內面，而 Tag 比較像內容的外面，所以 Tag 跟 keyword 是不相等的，Keyword 是這篇文章產生時就存在而不會改變，但 Tag 會因為使用者不同而產生不同的 Tag。

(二) 部落格

部落格 (網誌, Blog) 是近年來網路世界興起的另一股熱潮，它以網頁作為呈現媒介的個人專欄，也有人把它稱做網頁型態的專欄。從 1998 年起就在網路社群不斷的快速成長，且得到越來越多的重視與使用，1999 年 Peter Merholz 開始把 weblog 唸成 We Blog，因而有了 Blog 這個說法。

部落格可以說是個人或群體以時間順序所作的一種記錄，並且不斷更新。另一方面，部落格不只是將使用者的意見與想法傳遞在網路上，更可以讓興趣相投的網友交換建議並瞭解彼此[5]。

部落格之間的交流主要是通過回溯引用（TrackBack）和廣播/留言/評論的方式來進行的。一個部落格亦可被視為一個檔案，或是卷宗。與傳統檔案不同的是，部落格的作者既是這份檔案的創作人，也是其檔案管理人。而書寫部落格的人，通常稱為「blogger」（即部落格作者），既可以是一個人撰寫，也可以是一群寫。部落格一般採用了RSS技術，所有的RSS文件都必須符合由W3C發佈的XML 1.0規範。對讀者來說，可以通過RSS訂閱一個網誌，確知該網誌作者最近的更新。

（三）Ruby On Rails

Ruby 是一種簡單直觀、可擴展、可移植、物件導向的直譯式 scripting 程式語言。Rails 是以 Ruby 撰寫的完整（Full-stack）Web 開發框架，可以快速的開發 Web 應用程式，而且程式碼比以往更少是它的主要目的[1]。Ruby on Rails（ROR）的優點在於「簡潔與快速」，兼具 scripting 程式語言 PHP 的開發速度與易用性，以及 Java 結構式作法的「乾淨清楚」。這種組合加速程式的撰寫過程，日後的維護也容易[3]。

Rails 使用稱為 MVC 的應用程式架構，MVC 分別是模型（Model）、外觀（View）、控制器（Controller）的簡稱，可以將一個應用程式根據處理內容拆成不同的程式，簡化了應用程式的建立與修改工作，當維護某個部份時，可將整體的影響降至最低，也讓各部份的程式碼變的更容易閱讀，其三層式的 MVC 架構如圖 4 所示[1]。

長久以來，J2EE（Java 2 Platform, Enterprise Edition）不著重於讓程式簡單，而是複雜問題的解決。直到 Spring 這類輕量化開發框架的出現，開發工作變簡單，設計者容易設計出輕量化應用程式，又能勝任重量級任務的程式。這種組合加速程式的撰寫過程，日後的維護也容易。就像 AJAX 將桌面應用程式與傳統 Web 應用程式的優點結合在一起，ROR 融合 Java 與 PHP 這兩大極端，讓開發工作變得成既乾淨又俐落[8]。近來，有個很受歡迎的圖片分享社交性網站 WORDPHOTO（www.wordphoto.org）[8]。作者以 ROR 建造此 Web 應用程式，用了 40 個小時完成這個網站。看看 WORDPHOTO 這個網站，從註冊、增加、搜尋、評論、到添加標籤等，多數功能都是很豐富的；因為 ROR 直指：開發速度、簡潔、與生產力

的王道，提供一種更快速的開發方法，把精力放在創造想法和滿足需求上[8]。

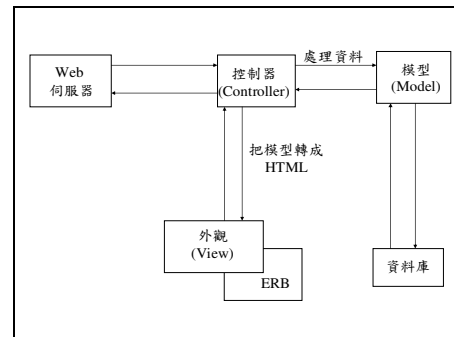


圖 4 Ruby on Rails 三層式的 MVC 架構[1]

（四）知識管理

由資訊科技之基本設施所建立而成的組織環境，充斥著現今知識主導的資訊社會，知識管理的核心問題是如何使資訊科技之基本設施，能夠對促成知識流通的環境作成有效的貢獻[4]。所謂的知識管理是指「適時的將正確的知識給予需要的成員，以幫助成員採取正確行動來增進組織績效的持續性過程。此過程包括知識的創造、確認、收集、分類儲存、分享與存取、使用及改進到淘汰等步驟」。所以知識管理是提供正確知識、幫助組織成員採取正確行動來增進組織績效的過程。另一方面，資訊是知識的輸入端，技術是知識的產出端，但資訊與技術的定義與內容較明確清楚，而知識是需要經由客觀分析與主觀認知形成，且與人相關，相對比較難系統化與明確的萃取；知識不同於資訊，其特徵在資訊要經過學習過程與價值認知方能形成知識；知識也不同於技術，技術是產品與服務的具體組成部份，因此僅屬於有形知識的一部份，而知識還包括產品與服務的抽象組成部份，並作為驅動技術創新與產品創新的重要基礎。

三、U-DO-EAT@Tainan 2.0 架構

本論文以 L.A.M.P 為系統開發平臺，並使用 ROR 為系統開發語言，依循 Web2.0 的概念透過 SupeSite 程式開發建置主系統 U-DO-EAT@Tainan 2.0[11]，並且包含兩個子系統，分別是府城知識地圖，以及府城小吃代理人來計算各項美食料理建議值的府城小吃推論系統，整個系統的架構如圖 6。

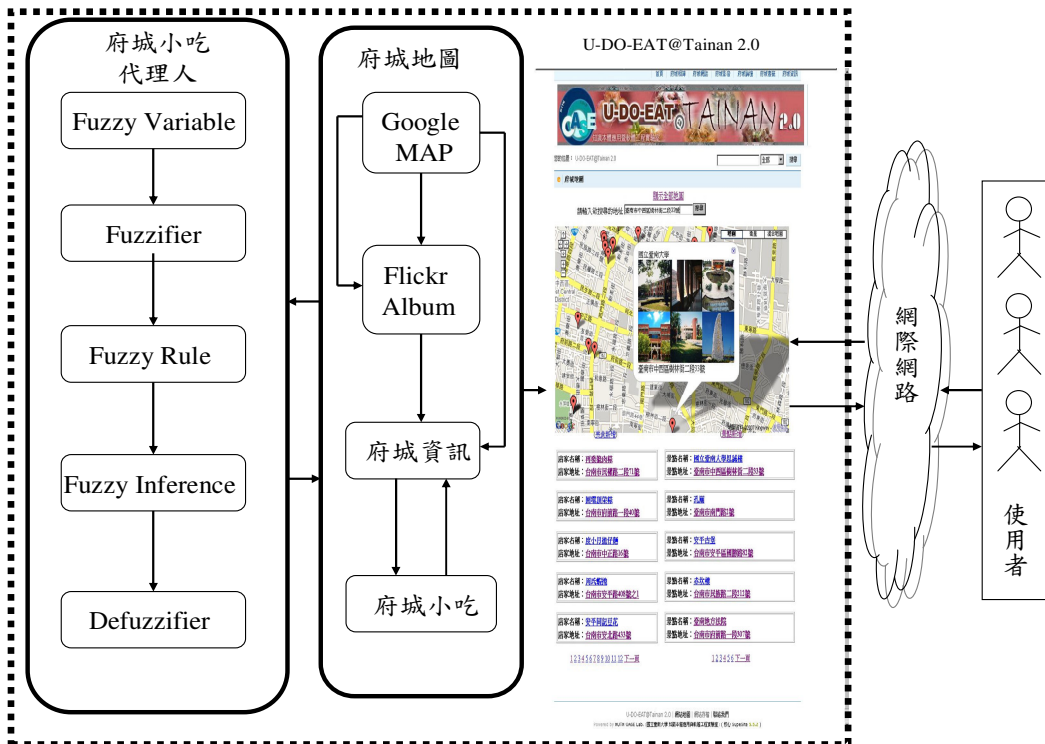


圖 6 U-DO-EAT@Tainan 2.0 系統架構圖

(一) 府城知識地圖

U-DO-EAT@Tainan 2.0 的子系統府城知識地圖架構如圖 5，除了具備傳統的美食網站的一般顯示出美食相關資料的功能以外，還有可以讓使用者自行新增相關詳細資料，系統會應用 Google Map 的 API，將詳細資料顯示給使用者以外，還包括了該美食的地理位置以及相關圖片等等，最後會將使用者所提供的新增資料將其儲存在資料庫裡，以供其它子系統運用，當使用者點選的其中一道食物資訊後，其地圖中心點便會移至該美食料理的位置，並且簡單的針對該項美食做介紹，如果網頁上並無使用者有興趣的資料，系統也提供地理譯碼，即使使用者只知路名或是目的地名稱也可以輕鬆取得相關資訊，無需再花費時間去尋找。

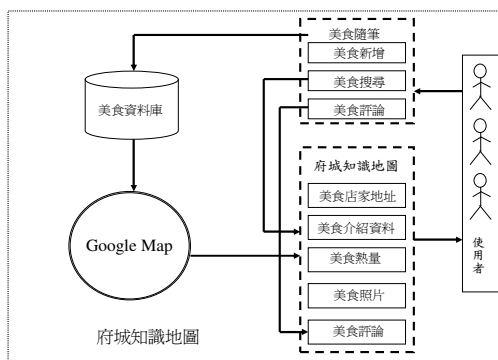


圖 5 府城知識地圖子系統架構圖

(二) 府城小吃代理人

府城小吃代理人應用模糊邏輯根據使用者的 BMI (Body Mass Index) 值、吃完小吃後所產生的熱量值以及小吃的口感油膩度透過模糊推論的機制推論出一建議值，提供給使用者參考該項美食是否建議食用，底下將簡要說明府城小吃代理人的模糊推論系統運作過程：

步驟1： 模糊化：共三個 Inputs 及一個 Output 分別為：

- 1、 使用者 BMI 值，範圍為 0 至 24。
- 2、 小吃口感油膩度，範圍為 0 至 1。
- 3、 小吃所產生的熱量，範圍為 0 至 1000，單位為大卡。
- 4、 給使用者的建議值，範圍為 0 至 1。

步驟2： 決定模糊集合

圖 7 為 BMI 值高低程度的隸屬函數 (Membership Function) 之圖示，圖 8 為小吃口感油膩度隸屬函數之圖示，圖 9 為熱量高低的隸屬函數之圖示，圖 10 為食物建議值的隸屬函數之圖示。

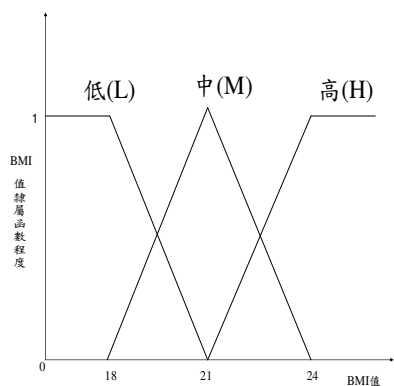


圖 7 使用者 BMI 值高低程度的隸屬函數

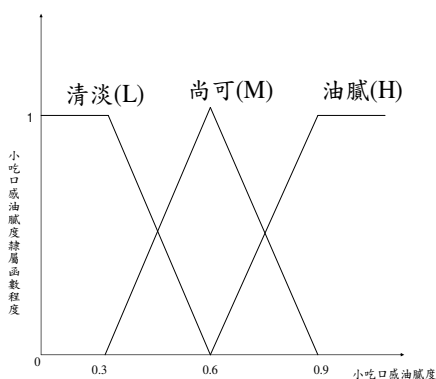


圖 8 小吃口感油膩程度的隸屬函數

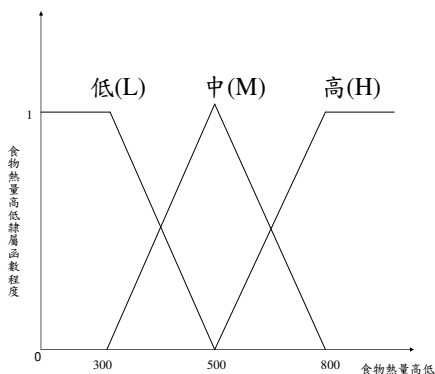


圖 9 熱量高低程度的隸屬函數

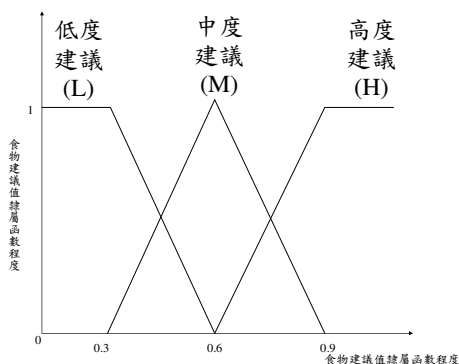


圖 10 食物建議值的隸屬函數

步驟3： 模糊法則

使用者 BMI 值、小吃口感油膩程度、小吃所產生的熱量以及給使用者的建議值四個模糊變數，三個輸入變數各有三個模糊數，共可得二十七條模糊規則，如表 1。

表 1 小吃建議值模糊規則

Rule	BMI	口感油膩程度	熱量高低程度	食物建議值
1.	L	L	L	M
2.	L	L	M	M
3.	L	L	H	L
4.	L	M	L	H
5.	L	M	M	M
6.	L	M	H	L
7.	L	H	L	L
8.	L	H	M	L
9.	L	H	H	L
10.	M	L	L	H
11.	M	L	M	M
12.	M	L	H	L
13.	M	M	L	H
14.	M	M	M	M
15.	M	M	H	L
16.	M	H	L	L
17.	M	H	M	L
18.	M	H	H	L
19.	H	L	L	H
20.	H	L	M	M
21.	H	L	H	L
22.	H	M	L	H
23.	H	M	M	M
24.	H	M	H	L
25.	H	H	L	M
26.	H	H	M	L
27.	H	H	H	L

步驟4： 模糊推論

由兩個部份組成，分別為 IF-part 和 THEN-part，也就是前提與結論，可表示為：IF < antecedent > THEN < consequent > 此僅列舉出三條為例：

- IF BMI 值 is 高 AND 油膩度 is 油膩 AND 熱量 is 高，THEN 建議值 is 低度建議。
- IF BMI 值 is 中 AND 油膩度 is 尚可 AND 熱量 is 中，THEN 建議值 is 中度建議。
- IF BMI 值 is 低 AND 油膩度 is 清淡 AND 熱量 is 低，THEN 建議值 is 高度建議。

並且根據匹配程度(Matching Degree)取min的運算，再藉由砍頭法(Clipping Method)得出推論結果，最後合成為可能解區域。

步驟5： 解模糊化

本論文用以解模糊化之方法為重心法(Center of Area, COA)藉由重心法對於合成後的區域進行解模糊化的運算，輸出最後的結果為一明確數值

(Crisp Value)，如圖 11 所示為模糊推論過程，圖 12 呈現以砍頭法為例得出推論結果以及將結果合成後進行解模糊化的執行程序。

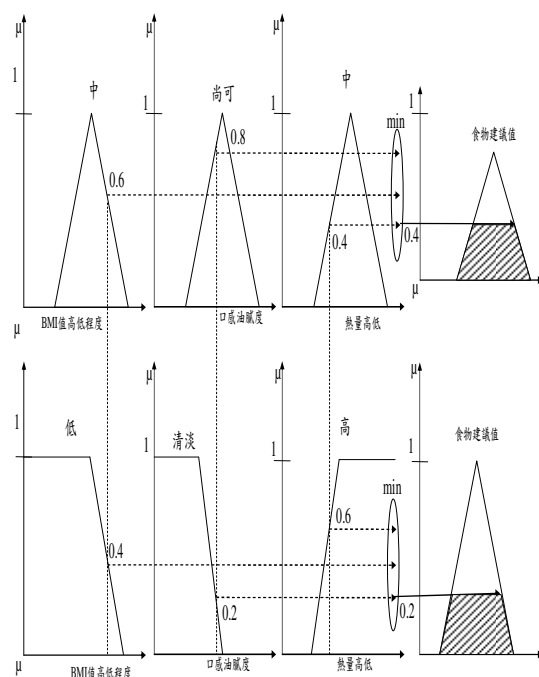


圖 11 模糊推論過程

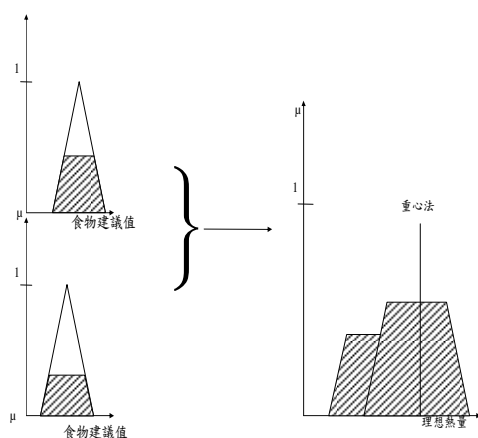


圖 12 重心法解模糊化

四 U-DO-EAT@Tainan 2.0 雛型

(一) U-DO-EAT@Tainan 2.0

圖 13 為 U-DO-EAT@Tainan 2.0 的主系統畫面(如圖 13)，依循 Web 2.0 的概念，系統給予每個註冊會員一個個人空間(如圖 14)，供註冊會員展

貼府城小吃相關的照片、影音以及網誌，使用者可以透過系統主頁面的上方功能表選擇府城相簿、府城網誌、府城影音、府城論壇以及府城書籤來瀏覽各項服務(如圖 15)。



圖 13 U-DO-EAT@Tainan 2.0 的主系統



圖 14 U-DO-EAT@Tainan 2.0 之使用者個人空間



圖 15 U-DO-EAT@Tainan 2.0 之府城相簿

(二) 府城知識地圖

當使用者點選的其中一道食物資訊後，上方的地圖便會帶至該美食料理的位置，並且可以點選食物名稱獲取該項美食的介紹(如圖 16)，如果網頁上並無使用者有興趣的資料，網頁上方也可以透過 Google MAP 的地理譯碼功能，即使使用者只知路名或是目的地名稱也可輕鬆取得相關資訊，無需再花費時間去尋找。



圖 16 小吃詳細介紹

另外府城知識地圖子系統亦提供了使用者自行新增美食資料，系統上所有的美食相關介紹資料以及照片將由使用者來建立，並且將其儲存至美食資料庫內，這部份的系統網頁畫面如圖 17；除此之外本系統更遵循著 Web 2.0 的概念，讓系統具備 RSS 以及 TAG 等等的功能。

府城地圖

店家名稱(*)：

店家食物(*)：

行政分區(*)：☐ 東區 ☐ 南區 ☐ 中西區 ☐ 北區 ☐ 安平區 ☐ 安南區

店家地址(*)：

店家電話(*)：

店家網頁：

描述內容(*)：

上傳圖片：

(註：*號者為必填)

圖 17 新增小吃及景點資料

(三) 府城小吃代理人

當使用者在府城知識地圖子系統瀏覽完美食資料後，首先，府城小吃代理人會將所選擇的項目顯示出來(如圖 18)，再透過府城小吃代理人勾選自身的條件需求，以及欲前往食用的小吃美食，系統將會根據表一的模糊推論法則，推論出建議值，並且透過 Google MAP 的 API 繪出府城知識地圖提供給使用者參考以及遊覽；並請使用者選擇其餘的使用者的 BMI 值、小吃口感油膩度以及小吃的熱量高低等三個模糊變數，再經過模糊推論後，府城小吃代理人將推薦使用者是否該食用該項小吃(如圖 19)。

