

友好程度推薦模組實作 – 以設群網站為例

張庭瑋

輔仁大學資訊工程學系

Email: 403226065@fju.edu.tw

范姜永益

輔仁大學資訊工程學系

Email: yyfanj@csie.fju.edu.tw

摘 要

隨著網路的興起，有愈來愈多使用者透過網路資源來尋找所需資訊，而在獲取資訊之眾多管道之中，又以推薦為較有效之方式，其原因在於通常需要推薦者缺少對該領域之知識，透過其他對於該領域有相關知識之人來給予建議，通常提問者可以得到較為正確之結果。因此，推薦系統在近幾年也順勢成為一個重要的議題，最常見之作法莫過於透過使用者評價之高低來作為推薦之依據。雖然當前最流行之推薦方法使用簡易且方便，但是卻缺少更精細之判斷—使用者偏好與回答者對使用者之理解。有鑒於此，本論文提出一推薦方法，以模擬社群網站 Facebook 之交流模式，並以紀錄使用者習慣以及回答者與提問者之間之親密程度為依據，做出更精確之推薦結果。

關鍵字：Facebook、社群網站、推薦系統

1. 前言

人們一生都有多到數不清的問題存在，隨著網路的普及，人們對於尋找知識的方式也跟著有巨大的轉變，從前人們必須從書面或者以口頭詢問的方式來獲取知識，到了網路發達的今天，只需要打開電腦，連上網際網路並且鍵入關鍵字或者在網站提出問題，很快便會得到許多答案，也已經成為了現代人最主要的資訊來源方式，而當人們需要在一個自己完全陌生的領域中得到正確的結果時，較常見之作法是詢問對於該領域有相關知識的人給予推薦，而隨著電腦網路的進步，目前已經有許多推薦系統給使用者使用，要得到推薦並不困難，但是由於提問者對於該領域缺乏知識[1]，難以對眾多推薦做出正確之判斷，加上回答者對提問者之喜好理解程度有所落差，進而影響提問者最後得到期望之結果，本論文提出之推薦方法，以模擬社群網站之模式，導入使用者習慣權重與好友之親密程度權重[2]，藉此在眾多推薦之中找出最適合提問者之推薦結果。

搭上網路科技進步的趨勢，網路已經成為許多人的資訊來源，越來越多人透過網路來獲取未知的知識，也是取得推薦方便且快速的管道，因此網路上出現不少推薦系統，替使用者從龐大的資料量中過濾資訊，只將使用者需要之資訊留給使用者，

但是在網路上使用者之間最缺乏的卻是互動關係。從前因為交流管道較少且較不便，因此人與人的交流通常為面對面交流而且朋友數量較少，因此較有機會深入理解對方之喜好，雖然能得到之推薦較少，較能給出符合使用者喜好之推薦，而在網路發達之後，人與人的交流雖然更便利了，尤其是社群網站，許多人動輒都能有上百甚至上千位朋友，但是其中有許多朋友通常缺乏交流，也缺乏相互之理解，雖然在提出問題後能讓使用者獲得大量之推薦，但是因為缺乏相互之理解，會出現過多難以過濾且無用之資訊，難以找出真正符合自身之推薦。綜合社群網站眾多使用者以及人與人互動觀察之優點，以模擬社群網站為例，透過統計使用者之間在社群網站上之互動多寡、使用者喜好[3]與網路上龐大資訊之綜合優點，在加以運算之後，便能從大量資訊之中找出最能符合提問者之推薦。

本論文對於現有推薦系統之貢獻主要是導入朋友分析機制。隨著社群網站日漸發達，人們交朋友的方式又多了一種。當今社群網站中，常常可見人人都有大量的好友，動輒上百位，其中有哪些是點頭之交，哪些是爾汝之交，哪些朋友真正瞭解當前使用者的興趣等等，都是容易混淆使用者吸收資訊的因素之一，因此在本論文中加入分析朋友的親密關係，藉此將過往直接接收推薦之方式加入新的考量方向並且加以修正，以便能將更正確的資訊過濾後傳遞給使用者。

2. 相關文獻

2.1 推薦系統

推薦之目的在於資訊交流，透過人們互相交流自身經驗或者相關知識來達到找出對特定的人之最佳解決方案。其中推薦系統分為傳統化與現代化兩類。

• 傳統化推薦系統

傳統化推薦系統主要包含”基於內容(Content-based filtering)”、“協同合作(Collaborative filtering)”與”混合推薦(Hybrid recommendation approaches)”等等，其共同點為推薦目標明確，較不易發生錯誤推薦方向之情況，例如：使用者 A 想買一支新手機，但是對手機卻非常陌生，因此某網路論壇發問，請其他使用者推薦該買

哪一支手機。但是其缺點為無法即時獲得當前環境之資訊，因此推薦準確度會受一定程度之影響。

- 現代化推薦系統

現代化推薦系統包含”本文感知 (Context-aware approaches)”、“基於語意 (Semantic based approaches)”與”基於跨領域 (Cross-domain based approaches)”等等。其特點在於可以獲取當前使用者所處之環境並且提出相對應之推薦，其推薦結果較能符合使用者之當前需求，但是缺點為較易發生錯誤推薦方向與需要事前建立好語意以及本體論系統。

2.2 親密關係程度分析

近年來社群網站盛行，而在社群網站上一定少不了好友這個部分，其中不難發現有很大比例的使用者有上百甚至上千個好友，但是這些朋友之中，通常只有少數是較有互動或者真正對該使用者較為瞭解的朋友，因此找出常有互動的朋友便成為一個推薦系統中的議題。透過分析友好程度，便可以在眾多資訊中找出真正符合兩使用者當前關係的資訊，藉此增加資料分析之準確度。

3. 情境設想

本系統以模擬 Facebook 之模式，以 Facebook 帳號登入並提供使用者留言與按讚之功能，其中，當使用者需要推薦與解答時，可以進入發問頁面提出問題，讓其他使用者來回答。設想：Ian 平時都會在 Facebook 上活動，並且在好友清單中 400 人。某天 Ian 想要買一台車，但是他對車卻非常外行，此時便可以透過本推薦方式 model 得到較能符合 Ian 的推薦。

- 一般貼文

Ian 平時每天都會在本網站中分享一些生活瑣事或者是有趣的事物，作為生活上之消遣，而在 Ian 的貼文底下，也都不時會有好友來回覆 Ian 的貼文，而 Ian 也會對其他好友之貼文做出評論或者回覆。

- 提問

Ian 利用 Facebook 帳號登入系統，而這個系統也是平常每天使用的社群網站，資料庫中存有 Ian 有興趣的頁面、與哪些使用者互動較為頻繁等資訊，而登入後便會看到提問與貼文兩個頁面，此時直接將問題提出，等待其他好友回答。

- 好友回答

在問題提出之後，Ian 的好友都會看到該問題，此時不論是誰都可以進入該問題頁面回答，在回答時，回答者必須選擇一個資

料庫中已經存在的喜好選項，當回答者所提出的答案不在資料庫時，系統便會要求使用者替該答案分類，方便未來其他使用者使用本系統。

- 判定推薦

在 Ian 的好友紛紛提出答案之後，系統會依據回答者與 Ian 的友好程度和回答答案是否與 Ian 興趣相符來排序答案，最符合 Ian 需求的答案將會被列在最上面。

4. 系統需求

依照上述使用情境，本寫節將敘述完整流程。

- 一般貼文。使用者登入網站，以 Facebook 登入並且驗證身份。成功登入後，使用者便可以直接貼文，在貼文成功建立後其他好友便可閱讀與回覆，同時也可以對其他使用者之貼文做出回覆。
- 提問。使用者登入網站，以 Facebook 登入並且驗證身份。成功登入後，使用者在貼文時選擇提出問題，問題成功建立後，便可以開始等待其他好友來回答。
- 好友回答。貼文者之好友都可以在文章列表內看到好友貼出之問題，可以直接進入問題進行回答。回答時，系統會直接讓使用者選擇推薦之項目，若推薦之項目尚未存在資料庫中，則系統會要求回答者幫忙輸入完整資料以建檔。
- 推薦判定。系統會將所有之回答項目做暫時之細部分離，並且放入另一模組等待計算。系統會依據被分離之部分做交叉比對、計算並且重新排序，計算完成之後會將分離之答案重組。系統會將所有推薦列出來，而顯示越上面之推薦為越符合使用者之推薦。

5. 系統架構

本推薦模型為一單獨運行子系統，屬於 Client-server 架構，但是可以套用於許多系統之上。Client side 主要任務為：“資訊呈現”、“資料讀取”，而 server side 任務則為：“儲存資料”、“資料運算”。

a. Client side

- 資訊呈現：本子系統提供網頁 UI 畫面呈現給使用者，方便使用者操作本系統
- 資料讀取：透過網頁 UI 介面，將使用者的輸入資訊與系統所需查詢，透過 Web service 傳送給 server side 並且進行運算。

b. Server side

- 儲存資料：在接收到來自 client side 的資訊之後，將其存入 server 上的資料庫，以方便日後存取。
- 資料運算：透過伺服器端的後端運算，以不斷更新資料庫中之資料，維持儲存之資料為

最新狀態。

其中，伺服器端以及客戶端中又可以細分為許多模組，以下將依序介紹：

5.1 客戶端模組

- 認證模組：Facebook login 模組，用來認證用戶身份，以提供相對應之服務。
- 貼文模組：提供給使用者發文與提出問題之模組。
- 回饋模組：透過回饋模組，使用者可以對好友貼文、問題以及感興趣的主題點讚，回傳喜好資訊給後端伺服器。
- 好友模組：本模組用來確認兩使用者之間好友關係。

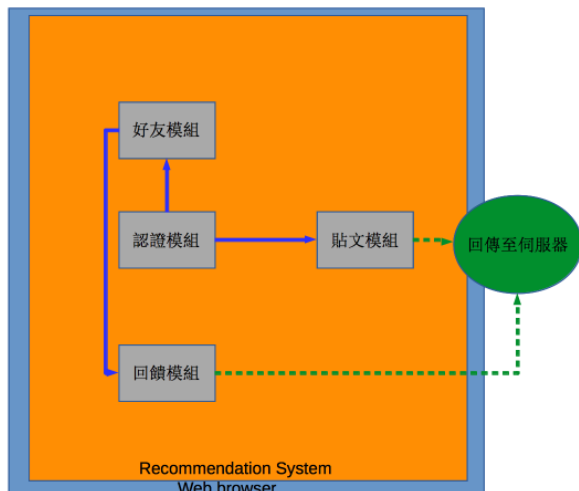


圖 1 客戶端模組圖

5.2 伺服器端模組設計

- 資料庫模組：用來處理資料儲存和前端與後端之連線。
- 用戶驗證模組：本模組用來認證登入系統使用者之身分，以及擷取系統所需之資訊。
- 文章管理模組：本模組用來管理所有使用者之貼文與提出問題。
- 文章回應模組：負責儲存每一篇文章與問題上之回應或者回答。
- 使用者關連模組：用於管理使用者之間之關連程度。
- 使用者喜好管理模組：管理使用者之喜好。
- 使用者行為管理模組：用於紀錄使用者之動作以及行為。
- 推薦模組：綜合問題提出者喜好與所有回答之關連，找出最適合問題提出者之回答。

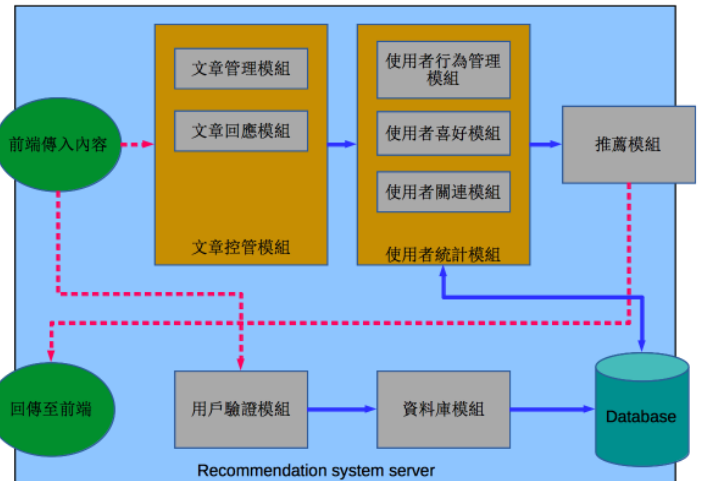


圖 2 伺服器端模組圖

5.3 循序圖分析

本小節將完整地以循序圖(圖 3)描述上述提及之模組實際在系統運行時之順序

1. 當使用者進入網站之後，系統會呼叫 `checkLoginState()`，先檢查是否有已經登入紀錄，若是沒有則呼叫 Facebook API 要求使用者使用 Facebook 帳號登入。在與 Facebook 取得身份認證後便會將使用者帳號、名字寫入資料庫、讀取好友資料並且轉跳至首頁，由於考慮到後續使用便利性以及安全性，在登入本網站後會產生 session 方便後續存取。
2. 當使用者貼文時(包含提問與一般貼文)，系統便會呼叫 `newPost()` 或者 `newAnswer()`，將該文章存入伺服器中並且將文章索引存入資料庫中。
3. 在貼文貼出後，便會在每位使用者看貼文時呼叫 `checkLike()`，確認讀者對貼文者之回饋狀態，同時將該狀態回傳至資料庫建檔。
4. 每當有使用者對問題提出推薦時，使用者統計模組就會呼叫 `passPost()`，將要進行推薦計算之問題傳入使用者統計模組之中，並且同時呼叫 `getRelationScore()` 與 `getUserLike()` 將親密程度與使用者喜好項目一起傳入使用者統計模組進行統整。
5. 當使用者統計模組完成統整後，最後的推薦模組會呼叫 `displayRec()`，此時會將使用者統計模組傳入之各種數值進行計算並且排序，最後呈現給提問者看。

6. 模組以及演算法

本章節將會細部討論推薦之方法以及推薦模組之詳細運作方式。

- 推薦模組

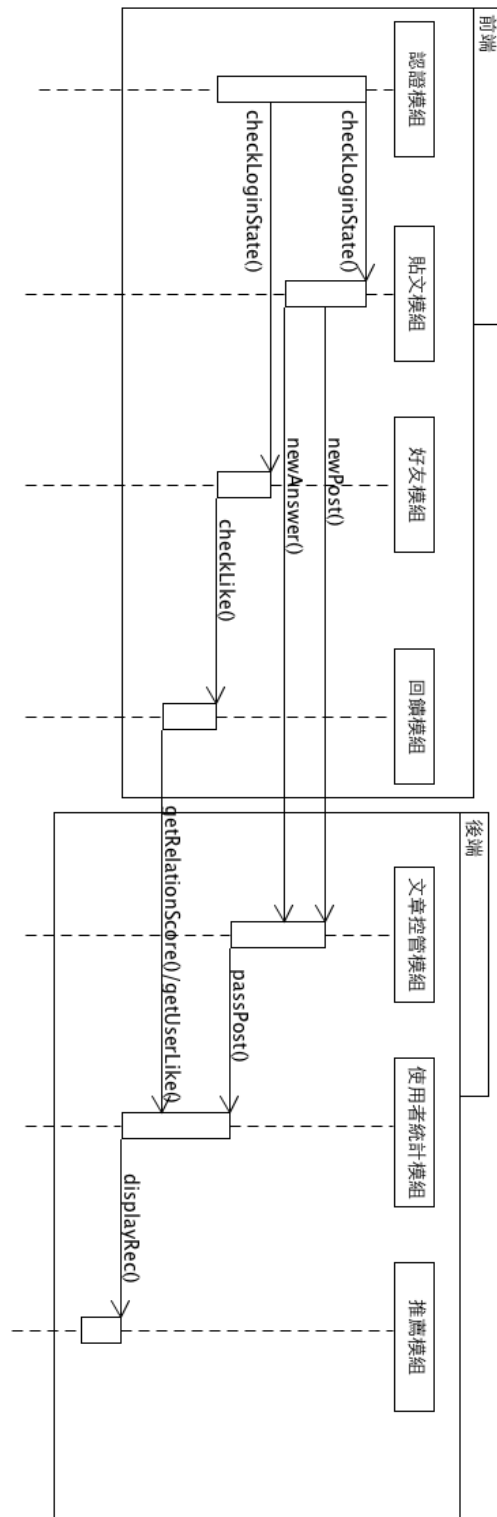


圖 3 系統運行循序圖

在問題出現回答之後，推薦模組就會被呼叫，在本模組中定義推薦權重 W_k 、且所有對於該問題之回答為 $R_F = \{r_1, r_2, \dots, r_n\}$ ，則一個問題底下之所有推薦為 W_{Rn} ， $W_{Rn} = (S_{Rn} \times 0.7) + (P_{mn} \times 0.3)$ ，而 $n = \{1, 2, \dots, n\}$ ，其中 S_R 為朋友友好程度而 P_m 為推薦相符程度，且 $P_{mn} = P_u \cap R_F n$ 。

回答依據最符合使用者提問者與否，由高到低進行排序，越上面的回答就是越符合使用者需求

之回答，而推薦模組在進行排序之前，則需要透過另外兩個模組先進行前制動作以擷取需要之數據。

• 使用者喜好管理模組

本模組最主要之功能在於管理使用者之喜好，在本系統中每一個喜好都會被歸類在一個分類之下，例如：Facebook 會被歸類在 Technology 之分類底下。定義使用者對每個類別的分數為 $SC = \{C_1, C_2, \dots, C_n\}$ 以及使用者有興趣之項目 $P_u = \{I_1, I_2, \dots, I_n\}$ ，而 I 為喜好項目且 $SC \rightarrow P_u$ 。每當使用者對某一喜好點讚時，本模組便會更新使用者之喜好。

在資料庫中，每一位使用者都會被記錄點讚過之喜好會被記錄下來，而同時也會記錄下使用者對每個分類之喜好程度，該程度是由分數來計算，初始為 0，每對一個喜好點讚，則該使用者的被點讚喜好之對應分類分數便會增加一分，例如：使用者對 Microsoft 點讚，而 Microsoft 的分類為 Technology，則該使用者 Technology 的分數就會加上一分，而 Microsoft 同時也會被紀錄為使用者有使用者

• 行為管理模組

本模組主要負責紀錄使用者在系統之中之行為，這裡的行為是指只使用者在系統之中的活動足跡，例如：回應、點讚等等行為，當使用者與其他使用者有任何互動時，本模組便會重新計算兩使用者之間之關係。

本模組中定義以下行為：點讚(C)、回覆(R)，其中 $C = 1$ 分 / 次而 $R = 2$ 分 / 次，一旦有任何回動，都會啟動本模組進行資料庫更新，例如：A 在 B 的貼文中回覆，則 A 與 B 之友好權重會增加兩分，數字越高代表兩人關係越好，有較大的機會更瞭解彼此。

7. 實驗結果與分析

7.1 伺服器端模組設計

本章將依照先前章節所提出之演算法，找四位使用者來進行實驗在模擬系統上進行實際操作。參與實驗之四位使用者之中，每一位都輪流擔任提問者並且提出一個問題，在實驗結束之後會請提問者填寫問卷，詢問其對於回答之滿意度。

表 1 為四位受測者有興趣之項目之資訊：

表 1 所有測試者喜好以及類別分數

使用者	喜好類別 / 類別分數
A	Asia / 1 Technology / 3 Dog / 1 Pop / 1 Rock / 1 Dota / 1 Japanese Animation / 1 Fantasy Novel / 1
B	Technology / 2 Pop / 2 FPS / 1 RTS / 1 Asia / 2 America / 1 Japanese Animation / 1
C	Technology / 3 Pop / 1 Dota / 1 RTS / 1 Asia / 2 Japanese Animation / 1
D	Technology / 3 Auto Mobile / 2 Asia / 2 Dog / 1 Motor Racing / 1 RPG / 1
E	Fantasy Novel / 1 Thrilling Movie / 1 Beauty / 1 Celebrity / 1 Pop / 1 Fashion Magazine / 2 American News / 1 Lingerie Brand / 1 Medical Drama / 1

表 2 為使用者之間之友好權重：

表 2 測試者友好權重

User \ User	A	B	C	D	E
A	X	13	19	200	32
B	13	X	308	119	10
C	19	308	X	273	12
D	200	119	273	X	50
E	32	10	12	50	X

以下表格為提出問題以及推薦，總共分為五種滿意度，分別為：非常滿意、滿意、中等、不滿意以及非常不滿意

提問者：A

提出問題簡述：推薦一本部無聊之小說

表 3 使用者 A 得到之推薦

排序	推薦人	推薦項目
1	D	盜墓筆記
2	C	魔法禁書目錄
3	B	治療
4	E	賈伯斯傳

推薦滿意度：非常滿意

提問者：B

提出問題簡述：推薦近期好玩之遊戲

表 4 使用者 B 得到之推薦

排序	推薦人	推薦項目
1	C	Tom Clancy' s: The Division
2	D	Grand Theft Auto 5
3	A	Clash of Clans
4	E	Outlast

推薦滿意度：非常滿意

提問者：C

提出問題簡述：推薦輔仁大學附近好吃之餐廳

表 5 使用者 C 得到之推薦

排序	推薦人	推薦項目
1	D	陶花園
2	B	麥當勞
3	A	名廚鐵板燒
4	E	漢口家

推薦滿意度：滿意

提問者：D

提出問題簡述：推薦近期好看的動作片

表 6 使用者 D 得到之推薦

排序	推薦人	推薦項目
1	D	蝙蝠俠對超人
2	A	美國隊長 3
3	C	自殺突擊隊
4	E	特工爺爺

推薦滿意度：滿意

提問者：E

提出問題簡述：請推薦台北好吃的美式餐廳

表 7 使用者 E 得到之推薦

排序	推薦人	推薦項目
1	D	TGI Fridays
2	A	雙聖
3	C	Toasteria Café
4	B	Campus Cafe

推薦滿意度：滿意

7.2 數據分析

在五位測試人員完成測試之後，我們提供了一份問卷給測試者填寫，並且透過統計圖表來檢視本推薦模組之可行性，而從以下圖(圖 3)表可知，五個測試人員中，40% 對於推薦結果非常滿意而 60% 則對推薦結果滿意，雖然沒有達到 100% 非常滿意，但是結果顯示透過友好程度來取得推薦能有效使測試者成功透過本系統之演算法找到適合自己之推薦結果，整體推薦之結果非常好，同時對本系統感到滿意。

請問您對於本次之推薦滿意度 (5 則回應)

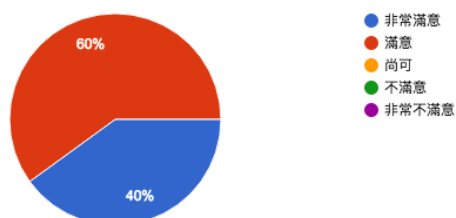


圖 3 測試者滿意度調查

8. 結論

網路的興起，改變了人們尋找資訊的模式，由書本、口耳相傳轉移到各個網站，同時，伴隨著社群網站的興起，社群網站也在無意間成為了人們尋求答案的地方。放眼望去，今天的人們已經與社群網站形影不離，不論是閒聊、分享或者是抒發情緒甚至是尋求解答，都能透過這個平台來完成，而透過網站中使用者之間的交流與各種動作，便可以發現使用者之偏好以及人際關係，當使用者在尋求答案時，透過先前蒐集之資訊，加以統計，便可以找出真正適合使用者之推薦，而本系統之終極目標則為替所有尋求答案之使用者找出最適合的推薦。本系統目前能蒐集之資訊有限，目前只限於留言和喜好項目，功能因技術之問題不盡完整，希望未來能做到直接從社群網站取得統計資訊、更詳細的使用者喜好以及更細微之使用者動作，並且加入推薦回饋之制度，加以確認推薦之正確程度，更能達到使用者所期望之更準確之推薦結果。

參考文獻

- [1] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, Grammati Pantziou, "Review Mobile recommender system in tourism", Journal of Network and Computer Applications 39 (2014), 319-333.
- [2] Antonio Moreno, Aida Valls, David Isern, Lucas Marin, Joan Borrás, "SigTur/E-Destination: Ontology-based personalized recommendation of Tourism and Leisure Activities", Engineering Applications of Artificial Intelligence 26 (2013), 633-651.
- [3] Frank E. Walter, Stefano Battiston, Frank Schweitzer, "A model of a Trust-based Recommendation System on a Social Network", Autonomous Agents and Multi-Agent Systems, 7 September 2007.
- [4] Takeshi Kurashima, Tomoharu Iwata, Go Irie, Ko Fujimura, "Travel Route Recommendation using Geotags in Photo Sharing Sites", 19th ACM international conference on Information and knowledge management (2010), 579-588.
- [5] Xuan Hau Pham, Jason J. Jung, Le Anh Vu, Seung-Bo Park "Exploiting Social Contexts For Movie Recommendation", Malaysian Journal of Computer Science (2014), Vol. 27 Issue 1, 68.