

輔助多人腦力激盪的智慧型代理人系統

Design of An Intelligent Agent for Stimulating Brainstorming

王俊翔

國立政治大學

資訊科學系

台北市，台灣

102703039@nccu.edu.tw

李蔡彥

國立政治大學

資訊科學系

台北市，台灣

li@nccu.edu.tw

摘要—近年來，腦力激盪討論逐漸成為團體決策所使用的主流方式，解決個別成員對特定主題缺乏想法的窘境，突破自身經驗的侷限性；然而，在討論過程中，由於成員組成不同，仍有內容不夠周全及進度停滯等問題。因此，本研究設計一個智慧型代理人系統，以機器學習方式建立特定專業領域的知識模型，在團體腦力激盪中，扮演一個虛擬成員，對過程中的對話進行自然語言處理，以了解目前的討論內容及判斷當下的討論進度，適時根據所建立的知識模型及回覆使用者對話模型，提出不同想法或延伸話題，增加團體成員討論的多樣性及全面性，並讓整個討論過程更加流暢。實驗結果顯示，我們的腦力激盪代理人系統對腦力激盪活動的進行，可以提供相當有效的導引與輔助。

Keywords—腦力激盪、智慧型代理人、機器學習、自然語言處理。

I. 介紹

腦力激盪解決部份傳統在團體討論上常遇到的瓶頸，但也可能因為缺少了一位有效的引導者，而造成效率不佳；因此，一位能調整流程及引導討論方向的成員便相當重要。本研究設計一個智慧型軟體代理人系統，加入腦力激盪過程，企圖在適當時機提出回覆，使得討論過程陷入膠著時，能重新引導團員們的思考方向，幫助腦力激盪的進行。

回顧目前的自然語言處理相關應用，常是以特定領域的資料作為系統背景知識，再針對使用者提出的句子，找尋背景知識中最相似的情況，以類似觸發事件的方式產生合適的回覆。例如，擷取網頁上大量的問題答案組合，所建構的心理層面問答系統[1]；以及利用維基百科上的資料，所設計的對話式智慧型教學系統[2]。然而，本研究設計的智慧型代理人系統不同於以往被動的角色，轉為主動者的

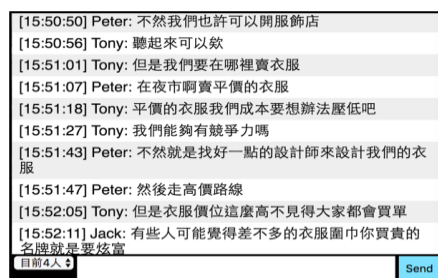
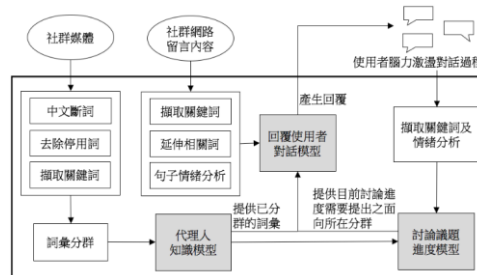


圖 2 系統操作畫面

身份參與對話；在參與腦力激盪時，並非針對語句與背景知識任一子句的相似度來觸發特定回應；而是在過程中，根據對話紀錄判斷目前討論走向，在合適時機主動產生回覆，由此來促進使用者們討論。另外，本研究選定以中文的腦力激盪流程為我們主要設計目標，故此智慧型代理人系統採用大量針對中文自然語言處理技術，並在腦力激盪流程的建構上有所著墨，實際使用畫面如圖 1。

II. 系統的三個模型

本研究設計智慧型代理人系統，加入多人腦力激盪的過程，輔佐多位使用者有效率的進行討論，並加強最後討論出之結果的完整性。整體系統架構圖如圖 2 所示，我們的智慧型代理人系統包含三個資訊模型（灰色方塊），包括代理人知識模型（Agent Knowledge Model, AKM）、討論議題的進度模型（Discussion Progress Model, DPM）及回覆使用者的對話模型（Response Generation Model, RGM）。



RGM)。

本研究擬定了一個腦力激盪討論方式通常會經歷的流程，如圖 3，並據以制定圖 4 的系統流程策略圖。從討論開場、說明討論目的，接著針對主題開始進行想法交流。在討論過程中，系統將整體流程劃分為兩階段，第一階段為話題發散，討論成員們會不停提出各自想到的面向，且在每個主題都先有初步討論、提出想法；第二階段為話題收斂，成員們會在先前已提出的想法中，找出有共識、可行的，進一步討論，若在途中還有其他異議，會再轉換至其他主題，不斷的循環這樣的過程，直至討論結束。本系統依據此擬定的流程訂定討論策略，在參與腦力激盪途中，引導成員們能夠更有效率的完成腦力激盪。

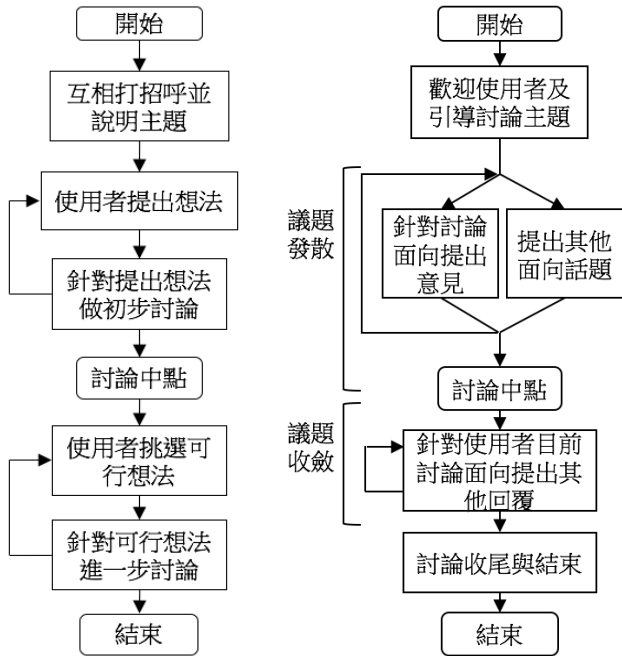


圖 4 典型腦力激盪流程圖

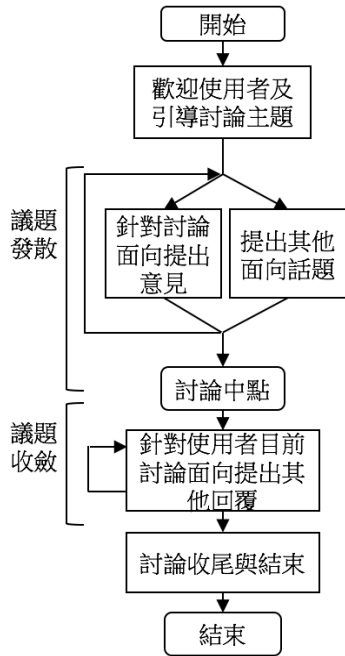


圖 5 本系統流程策略圖

$$D_i = \{P_1|P_2|\dots|P_t\} \text{ for } i = 1 \text{ to } N \quad (1)$$

$$S_{r_i}^i = \text{TextRank}(P_{r_i}) \text{ for } P_{r_i} \text{ in } D_i \quad (2)$$

$$D'_i = \{S_{r_i}^i | S_{r_i}^i \text{ is the top 10\% in } D_i\} \text{ for } i = 1 \text{ to } N \quad (3)$$

$$D_c = D'_1 \cup D'_2 \cup \dots \cup D'_N \quad (4)$$

$$T_s = \{P_i | \text{kmeans}(P_i) \in s\} \text{ for } P_i \text{ in } D_c \quad (5)$$

N : 文章總數、 t : 詞彙總數、 s : 分群標籤

A. 代理人知識模型 (AKM)

代理人知識模型事先收集大量常見到的社群媒體（如：Ptt、Wikipedia）上的中文文章內容，針對所有文章進行斷詞、去除停用詞後，以多個詞彙 \mathcal{P} 的集合表示每一篇文章內容 \mathcal{D} (1)；接著，本研究使用語言模型 TextRank[3]，評估在同一篇文章中每個詞彙所具備的重要性分數 S (2)；再將排名前 10% 的詞彙視為能夠代表文章的關鍵詞，並且擷取出各篇文章中所有關鍵詞，組成該文章內容新的表示方法 \mathcal{D}' ，同(3)。最後，把各篇文章新表示方法中的所有詞彙，聯合形成此模型的知識語料庫 \mathcal{D}_c (4)。並採用 K-Means 演算法[4]，將語料庫裡的所有關鍵詞依照主題分群，得到各主題的詞彙集合 \mathcal{T} (5)。我們將分群的結果及各群所涵蓋的詞彙存進資料庫，以作為代理人的知識模型。

B. 討論議題的進度模型 (DPM)

本研究根據所設計的腦力激盪各階段所要採取回覆策略，如圖 4，來建立討論議題的進度模型，全程以規則模式的方式將討論進度模組化。此模型在腦力激盪過程即時紀錄全部對話序列 \mathcal{J} ，並將每一句話依照出現順序編號(6)；且選定一定數量 r 以內的對話內容，視為近期討論內容序列 \mathcal{R} (7)。針對這兩個序列，利用代理人知識模型分別計算對話中所含各主題標籤的次數(8)(9)，並將全部對話序列計算出的標籤結果 $\mathcal{L}_{\mathcal{J}s}$ 作為累積標籤次數；近期對話序列

的標籤結果 $\mathcal{L}_{\mathcal{R}s}$ 則扮演近期討論走向的議題組成。對於議題發散及議題收斂的過程分別套用不同規則來判斷。

$$\mathcal{J} = \mathcal{J}_1 \cdot \mathcal{J}_2 \cdot \dots \cdot \mathcal{J}_i \quad (6)$$

$$\mathcal{R} = \mathcal{J}_{i-r+1} \cdot \mathcal{J}_{i-r+2} \cdot \dots \cdot \mathcal{J}_i \quad (7)$$

$$\mathcal{L}_{\mathcal{J}s} = |\{i | \text{AKM}(i) \in s\}| \text{ for } i \text{ in } \mathcal{J} \quad (8)$$

$$\mathcal{L}_{\mathcal{R}s} = |\{i | \text{AKM}(i) \in s\}| \text{ for } i \text{ in } \mathcal{R} \quad (9)$$

s : 分群標籤

處理發散階段的程序如 Alg.1，設定一門檻 τ_1 ，本模型會將近期討論議題組成排序後，依序確認討論主題的累積標籤次數是否在門檻 τ_1 內，來決定延續目前主題或是認定討論足夠該提出其他面向。當成員們近期關注到的主題，累積討論數量皆達到門檻時，代表之後順位的話題在最近都未提出相關討論，所以會從目前討論議題以外的話題中隨機選出一個，這時系統回覆的目的是嘗試發散議題，此模型會利用提出其他面向的內容來刺激使用者們更多元的討論。最後，當所有主題的累積討論數量皆大於門檻 τ_1 時，此模型會認為目前在發散階段的討論都已足夠，進入討論中點，結束議題發散的過程，並開始進行議題收斂。

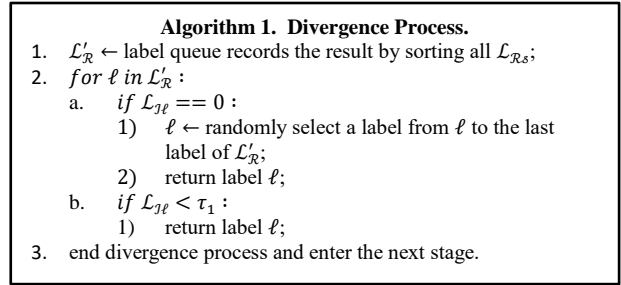


圖 5 模型 DPM 於發散過程演算法

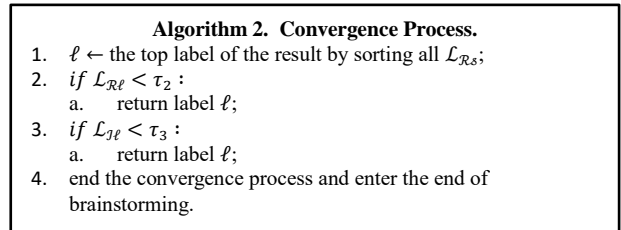


圖 6 模型 DPM 於收斂過程演算法

議題收斂的過程如 Alg.2，設立兩個門檻 τ_2 、 τ_3 來判定進度： τ_2 是跟近期的標籤數量比較， τ_3 則是與累積討論數量比較。在收斂過程的主要策略是不斷提出有關目前討論走向的內容，直到討論程度被認定足夠。本模型會取得近期討論最熱烈的主題當作目前走向，並且判斷此主題在近期及累積提到的次數是否分別已經抵達兩個門檻。滿足兩個門檻後，此模型會認為使用者們目前的討論走向在近期及整體兩方面的討論程度都已足夠，因此進入收尾階段；反之，若有一個門檻尚未達到，即會延續使用者當前的討論，讓使用者對於議題能夠有更深入交流。

另外，此模型也會在兩階段過程中，依據使用者的對話統計，判定刺激氣氛的時機，以要求特定使用者提出意

見或針對當前內容詢問是否有相關想法等回覆，促進所有人都能提出自身看法，熱絡討論的進行。

C. 回覆使用者的對話模型 (RGM)

本模型是用來產生要回覆的內容，針對當下處於發散、收斂時機及要回覆的主題面向，找尋最適當的句子。事先對於要討論的特定議題，收集社群網路的留言內容（如：Dcard）；將每則留言訊息 \mathcal{M} 斷詞分成數個詞彙 \mathcal{P} (10)，並使用 TextRank 模型[3]，計算每個詞彙在該留言中的重要性 \mathcal{S} (11)，並將分數最高者選為該句的關鍵詞 \mathcal{K} (12)；接著，留言內的關鍵詞會挖空存成新的句子 \mathcal{M}' (13)。另外，利用維基百科的中文文章內容當作資料集，以 Word2vec 演算法[5]訓練出神經網路模型，計算詞向量後，取出向量與各個留言的關鍵詞距離於一定界線 ℓ 內的詞彙(14)，認定那些詞彙與關鍵詞相關，將其存成該句留言的關鍵詞延伸詞彙集合 \mathcal{K}' (15)。本模型也利用 Naive Bayesian 的分類演算法[6]，針對網路取得的評價資料訓練情緒分析模型，再套用於所有收集到的留言內容，得到該句話所隱含的正負面情緒數值 \mathcal{N} (16)。最後，每個挖空的句子、延伸關鍵詞的詞彙集合及句子的情緒值，會存成組合作為延伸話題的回覆使用者句子清單 \mathcal{C} (17)。

$$\mathcal{M}_i = \mathcal{P}_1 \cdot \mathcal{P}_2 \cdot \dots \cdot \mathcal{P}_t \quad (10)$$

$$\mathcal{S}_k^i = \text{TextRank}(\mathcal{P}_k) \text{ for } \mathcal{P}_k \text{ in } \mathcal{M}_i \quad (11)$$

$$\mathcal{K}_i = \mathcal{P}_k \text{ for } \mathcal{S}_k^i \text{ is the top in } \mathcal{M}_i \quad (12)$$

$$\mathcal{M}'_i = \mathcal{P}_1 \cdot \mathcal{P}_2 \cdot \mathcal{P}_k \cdot \dots \cdot \mathcal{P}_t \text{ let } \mathcal{P}_k = e \quad (13)$$

where $\mathcal{P}_k = \mathcal{K}_i$ and $1 \leq k \leq t$,
e: sign for empty mark

$$\mathcal{E} = \{\mathcal{P} \mid |wTv(\mathcal{P}) - wTv(\mathcal{K}_i)| < \ell\} \quad (14)$$

wTv: word2vec、 ℓ : 距離界線

$$\mathcal{K}'_i = \{\mathcal{K}_i\} \cup \mathcal{E} \quad (15)$$

$$\mathcal{N}_i = \text{NaiveBayes}(\mathcal{M}_i) \quad (16)$$

$$\mathcal{C} = \{(\mathcal{M}'_i \mid \mathcal{K}'_i \mid \mathcal{N}_i)\} \quad (17)$$

$$\mathcal{R}' = \mathcal{J}_{i-s+1} \cdot \mathcal{J}_{i-s+2} \cdot \dots \cdot \mathcal{J}_i \text{ where } s < r \quad (18)$$

$$\mathcal{N}' = \text{AVG}(\text{NaiveBayes}(\mathcal{J}_k)) \text{ for } \mathcal{J}_k \text{ in } \mathcal{R}' \quad (19)$$

模型也以人工方式建立了多組特定情境的句子清單，如開場時「大家都可以開始了嗎」、用於刺激討論氣氛的「還有沒有其他的可能呢」等，直接以整句句回覆，來增進使用者的體驗。其中人工建立用於發散話題的句子清單，「如果是__這方面呢」等，使用時找到目標主題的詞彙直接填入開放性問題的句子，形成「如果是機器人這方面呢」句子，回覆至聊天室，刺激整體思考方向改變，讓討論成員能夠有機會轉往該面向進一步提出相關想法。

在腦力激盪過程中，此模型會針對不同處境實施不同的回覆產生機制，如 Alg.3。根據討論議題的進度模型所分析的結果，得到目前的目標主題及討論當下要採取的回覆策略。如果現在處於要使用特別情境的對話清單，模型會從該清單中隨機挑選句子回覆至聊天室；然而，若現在的情況是要提出其他面向的主題來刺激成員的思考，則會

Algorithm 3. Response Generation.

1. $\ell, s \leftarrow$ get the label and current situation from DPM;
2. if s is not for continuation and not for divergence :
 - a. return a response from the specified set \mathcal{A}_k randomly;
3. if s is for divergence :
 - a. $t \leftarrow$ get a term from the target ℓ cluster randomly;
 - b. combine a sentence from the set \mathcal{A}_d with t and return;
4. if s is for continuation :
 - a. accumulate the average emotion value \mathcal{N}' from \mathcal{R}' ;
 - b. $k \leftarrow$ get the keyword from \mathcal{R}' ;
 - c. if k is in ℓ cluster :
 - 1) $t \leftarrow k$;
 - d. else :
 - 1) $t \leftarrow$ get a term from the target ℓ cluster randomly;
 - e. $r \leftarrow \{(\mathcal{M}'_i \mid \mathcal{K}'_i \mid \mathcal{N}_i) \mid t \text{ in } \mathcal{K}'_i\}$ for all 3-tuple in \mathcal{C} ;
 - f. $\mathcal{M}', d \leftarrow$ get \mathcal{M}'_i and the distance by finding the least distance $|\mathcal{N}_i - \mathcal{N}'|$ from r ;
 - g. if $d < \tau_4$:
 - 1) combine \mathcal{M}'_i with t and return;
 - h. else :
 - 1) pass this time.

圖 7 模型 RGM 回覆產生演算法

利用代理人知識模型找出目標主題的詞彙，來與人工建立的開放性問題清單組合，產生回覆；另外，在延伸成員目前討論話題的部分，此模型如同討論議題的進度模型，也用到近期討論內容序列，並設定一個情緒參考點 s ，從序列中取出更接近當下的 s 句話 \mathcal{R}' (18)，經過 Naive Bayesian 分別預測每一句的情緒數值後取平均數(19)，視為當下正負面情緒值 \mathcal{N}' ，作為之後排序恰當回覆的標準；也會從此對話序列中找出關鍵詞，來作為最優先的可能詞彙，確認該關鍵詞符合目前應回覆的目標分群，若否，則隨機選取目標分群的詞彙。選定詞彙後，從回覆使用者句子清單，找出延伸詞彙集合所有涵蓋該詞之組合，計算出句子情緒值與當下情緒值差距最小者作為候選回覆。為了避免選定回覆的情緒值與當下情緒值差距過大，而導致使用者感到過於突兀，另外設立一情緒門檻 τ_4 ，確認情緒值差異是否在範圍之內。

III. 系統評估

A. 系統實作

本系統目前設定的討論主軸為「創業想法的產生」。在代理人知識模型方面，收集中文社群媒體批踢踢實業坊 (Ptt) 及維基百科 (Wikipedia) 涉及到各個產業的文章內容，針對文章進行中文斷詞；應用 TextRank[3]語言模型來擷取文章內的關鍵詞，並操作 K-Means 演算法[4]，將所有取得的關鍵詞，分為七個主題集合，作為系統的背景知識。討論議題的進度模型方面，實作本研究提出兩個演算法 Alg. 1 及 Alg. 2，分別在不同的討論時機，判斷所要採取的回覆策略。另外，回覆使用者的對話模型，本系統從社群 Dcard 網站上與產業相關文章的留言內容，以如同代理人知識模型所採用的方法來擷取關鍵詞；並以維基百科網站的文章內容作為訓練資料，計算中文詞彙的詞向量[5]，建立神經網路模型後，利用詞向量距離，針對句子內關鍵詞

延伸出其他可能相關詞；此模型也採用 Naïve Bayesian 分類演算法[6]，建立情緒分析模型，套用於留言句子內；最後，實作組合回覆的過程 Alg. 3，產生代理人的回覆句子。

B. 實驗設定

本研究利用自行設計的網頁式聊天室作為實驗介面。總共進行四次實驗，在每次實驗中，我們均邀請三位受測者，在不知道彼此身份的情況下，連同此智慧型代理人系統，總共四位討論成員，一同遠端連接聊天室，以匿名編號的方式交談，以完成腦力激盪討論。

在整體腦力激盪討論結束之後，我們會讓每位使用者填寫兩階段式的問卷，受測者會依據三個面向：整體發言表現、延伸目前話題表現、提出新討論面向的表現，以零分至滿分十分整數級距做評分。而第一階段的問卷是讓受測者對於每個面向，依序針對其他三位討論成員評分；第二階段，則是會說明討論過程中含有一位成員為智慧型代理人系統所扮演，請受測者依據談話內容，猜測系統所扮演的成員編號為何，並且對它再次依據三個面向評分；最後本研究將收集到的問卷分數，依據三個視角計算平均值並且顯示為圖表分析：系統外的成員互評、成員在未知有代理人系統參與的狀況下對系統的評分，以及在已知有一系統的情況下對該系統評分。

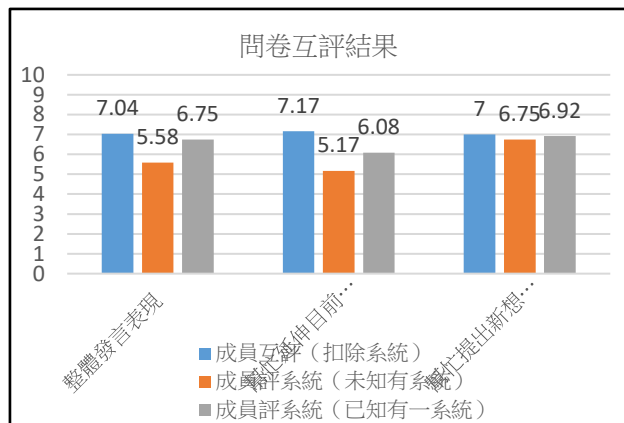


圖 8 實驗結果

C. 實驗結果

實驗結果統計如圖 8，在成員未知有系統時，系統在幫忙提出新想法的表現為 6.75 分較延伸話題優秀，且已經非常接近其他成員互評的平均分數 7 分，顯現出有實質的幫助。由結果推估是因為在發散話題的過程，系統選定分群裡的詞彙直接填入人工建立的句子清單回覆，如此的組成方式，只要能夠有效計算尚未提出話題的分群，選出可能回答的詞彙，回覆就不容易有太多的偏差。反之，在幫忙延伸話題方面，可歸因於系統目前僅有從情緒值篩選回覆語句，仍然容易因上下文語氣差異，造成談話不連貫，因此分數 5.17 分與成員互評分數 7.17 分有明顯差距。

另外，雖然在第二階段問卷中，請受測者猜測何者為系統所扮演角色的答對率為 100%，但是在受測者知道其

談話內容全為系統自動產生後，受測者對於系統在各方面的表現皆為更加肯定，尤其在延伸話題方面，成員在重新審視過程中的對話後，給予更正面的回饋，分數增加至 6.08 分，提升了 0.91 分；在整體發言的表現上，成員們知道代理人系統身份後給予的分數上升為 6.75 分，十分接近其他成員互評的 7.04 分，因此可知道討論成員們十足肯定系統的參與。

客觀上分析四次實驗的腦力激盪過程。從討論開始至抵達討論中點，四次實驗分別經過 136、146、149、191 句對話。審視較突出的 191 句討論內容，發現原因是成員們提出的句子，超出了系統的知識背景，因此無法有效累積各主題的標籤次數。在這過程裡，系統也不斷引導成員將話題導向系統所涉獵的主題。另外，整體流程至結束所經過的對話分別是 192、246、234、293 句，本研究觀察花費最少句的對話過程，在話題收斂的階段，討論成員們都很集中的討論其中兩個主題，沒有太多的話題轉換，因此能夠快速地累積分群標籤次數，抵達討論結尾。

IV. 結論

本研究設計的智慧型代理人系統不同於以往問答系統的應用，扮演主動者的身份參與腦力激盪的過程，是在自然語言處理領域的一創新應用。本研究的系統能有效利用我們所建構的代理人知識模型及討論議題的進度模型來判斷目前討論的狀況，並能依據當下採取的不同策略，以回覆使用者的對話模型提出符合的語句，來引導議題面向的增加，刺激使用者在更多話題上的想法激盪，以及熱絡討論過程的氣氛。

由於目前本研究的智慧型代理人系統回覆使用者的對話模型中，並沒有太多的語意分析，導致有時回覆的句子裡，關鍵詞是符合的，但卻缺乏與對話紀錄過程的上下文語氣關係的連結，以致使用者在討論過程，雖然能明白本智慧型代理人系統所要提及的議題，但卻會感到有些突兀。未來，我們將會持續增進本智慧型代理人系統在語意分析上的技術處理，包括使用者談話內容的處理及回覆使用者的對話模型，企圖讓本系統提出的語句更加符合當下的情境，以利增進使用者體驗。

REFERENCES

- [1] Y. Liu, M. Liu, X. Wang, L. Wang and J. Li, "PAL: A Chatterbot System for Answering Domain-specific Questions," ACL, 2013.
- [2] K. Lee, P.H. Seo, J. Choi, S. Koo and G.G. Lee, "Conversational Knowledge Teaching Agent that Uses a Knowledge Base," 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2015.
- [3] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," ACL, 2004.
- [4] K. Wagstaff, C. Cardie, S. Rogers and S. Schroedl, "Constrained K-means Clustering with Background Knowledge," ICML. Vol. 1, 2001.
- [5] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient Estimation of Word Representations in Vector Space," arXiv preprint arXiv:1301.3781, 2013.
- [6] A. Go, R. Bhayani and L. Huang, "Twitter Sentiment Classification using Distant Supervision," CS224N Project Report, Stanford 1.12, 2009.