以知識本體為基礎建構病毒分類知識庫系統

摘要

近年來軟體技術的成熟發展，讓網路上充滿著許多樣式的電腦病毒，而這些威脅數量也不斷的在攀升當中，若使用者不幸遭受到網路上的電腦病毒感染與威脅，就應該儘快識別出電腦病毒的型態，並找尋相關解決方案。則要防範遭受到網路上電腦病毒的感染與威脅，本研究提出藉由知識本體結合機器學習的方式，進行電腦病毒領域的偵測與查詢，知識工程師收集電腦病毒特徵後，經由電腦仿造人類學習的模式便可產生電腦病毒的知識階層架構，此方法可以迅速且便利的塑模知識本體的知識架構，則知識工程師參照此階層架構便能建置此領域的知識本體，最後利用知識本體的優點，推論出電腦病毒間隱含的關係，並且提供相關的解決方案給予使用者。

關鍵詞：電腦病毒、知識本體、機器學習、知識架構

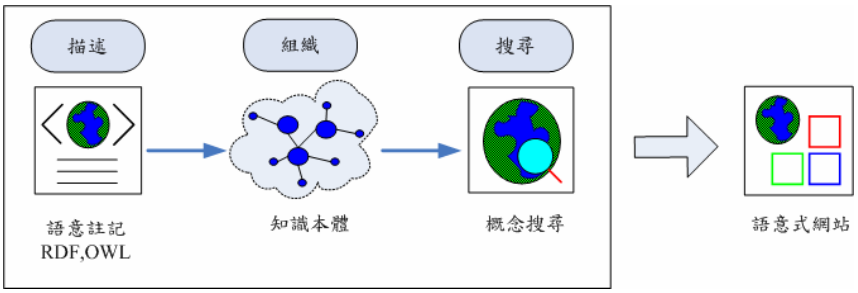
壹、緒論

語意網(Semantic Web)是由全球資訊網路創始人 Berners-Lee (2001)在研究中所提出，研究中則宣告未來網際網路的發展趨勢。語意網與全球資訊網路的不同點在於它能區別出詞彙(Vocabulary)的意義，且利用知識本體(Ontology)的知識架構能正確判斷詞義，知識本體也可以進行推論和訊息整合等能力，因此使用網路上的語意規則與知識本體，將分散各處的訊息結合，並且正確的找出搜尋結果。

Swartout 與 Austin(1999)在研究中提到「知識本體可以作爲知識表達的基礎，避免相同的領域知識被重複的分析，並且有統一的術語和概念去實現知識共享的目的。」，則此論點將可被具體的實踐在語意網路中。Maedch 與 Staab(2001)也提到「語意網在很大的程度上依賴著知識本體的結構，將這些資訊轉換成機器可理解的訊息。」，目前所發展出來的語意網路是利用資源描述框架(Resource Description Framework, RDF)和通用資源標誌碼(Universal Resource Identifier, URI)連結網頁相關資源，接著經由超連結找到關鍵詞，最後使用知識本體定義相關關鍵詞，並做邏輯規則上的推理運算，如圖一所示。

一般來說，知識本體應用於知識庫系統之建構，可分爲知識本體的建置及規

則推論的應用(Chi, 2008)。知識本體的建置程序有三項：知識擷取、知識塑模及知識表達(Noy, 1997; Uschold, 1996)，本研究主軸將關注於電腦病毒領域工程師所收集到的病毒特徵，轉換成爲機器可讀的知識架構，提出利用機器學習的方式，自動化產生知識本體的病毒知識架構，解決知識工程師在建置病毒知識分類架構時的困難和降低建置時間。現今知識本體在電腦科學的領域上應用非常廣泛，例如知識工程、知識表達、語言工程、資訊塑模、資訊檢索等(Guarino, 1998)，在這些特定領域之中，知識本體扮演著知識模型的呈現與表達知識間關係的角色，有鑑於此知識本體的分類架構品質就顯的非常重要。隨著網際網路的使用率增加，大量的資訊和文件也跟著相繼湧出，若以人工的方式去分類知識，將會顯的非常困難且耗時，近年來許多知識分類的方式相繼被提出，例如圖形化方式、資訊檢索方式、機械學習法則等。



圖一 語意式網站 (資料來源：戚玉樑，2006)

本研究針對電腦病毒領域提出自動化分類知識架構的方法，由於在電腦病毒的知識來源則需擷取病毒專家的經驗或知識，因此電腦病毒的知識塑模將是以專家知識作爲塑模之依據，當知識工程師不具有此領域的背景知識，或者知識工程師遭到替換時，再次塑模知識本體的過程將會變得艱辛且耗費時間，則所產生的知識本體不具有彈性擴充的空間或欠缺完整性的關聯；其次，電腦病毒的型態不斷的演變，傳統的電腦病毒分類架構，使得使用者越來越難利用過去電腦病毒的定義去劃分這些惡意程式的類型，如電腦蠕蟲、特洛伊木馬等，在每個分類架構中，各有數千種以上變種的電腦病毒，目前的分類架構並不能清楚且明確的去辨識某特定電腦病毒屬於何種分類架構，相繼造成使用者在病毒資料庫搜尋上的不便。根據前述的問題，本研究將提出自動化建構知識本體分類架構的方法，解決傳統知識工程師塑模時，需要依照專家經驗知識所定義的知識架構，如此將會降低整體知識架構的完整性，其次，增加知識分類架構的擴充性，知識工程師只需要將所擷取出的知識特徵，經過機器學習法則的反覆訓練與學習，便可產生出知識分類架構，知識工程師即可參照此分類架構塑模知識本體，此做法將能隨著時間增長變動擴增知識分類的架構，也降低知識工程師的負擔。

本研究於第二節針對電腦病毒、知識本體及機器學習進行研究之回顧；第三節將介紹本研究所提出之系統架構；第四節爲知識本體應用於電腦病毒之實作；最後，提出本研究之結論。

貳、文獻探討

本研究以電腦病毒分類架構建置知識本體之知識庫系統，首先第一節會對電腦病毒做個概括性的探討，第二節介紹知識本體領域相關技術、知識本體特性，第三節爲機器學習法則中的自我映射組織模型和支援向量機模型理論基礎介紹。

一、電腦病毒 (*Computer Virus*)

近年來因爲網路的興盛，造成許多嚴重的議題發生，如非法入侵、阻絕式服務攻擊、及電腦病毒的出現，特別是電腦病毒的攻擊能造成大量電腦系統的癱瘓(Shih, 2005)。電腦病毒由Cohen (1987)在論文中首次提出，文章中明確定義電腦病毒爲一種會不斷「自我複製」及「感染」的程式，並描述作者與其他專家對電腦病毒研究的實驗成果。電腦病毒概念類似於生物學上病毒的特性，它具有自我複製、感染及破壞的能力，當透過網路入侵於使用者的電腦後，將會危害電腦系統整體上的運作，且針對系統內部進行各種破壞，導致使用者有一定程度上的威脅與損失。

現今軟體技術的成熟發展，讓網路上充滿著許多樣式的病毒，而這數量也一直在不斷的攀升，然而目前各家防毒軟體廠商也對電腦病毒有著不同的區分，譬如趨勢科技公司針對2001年所爆發的Code Red 電腦病毒，其後定義駭客型態的病毒分類，而賽門鐵克公司也對近年網路上的威脅新增惡意程式碼的分類，雖然電腦病毒近年一直不斷的演進與增加，但基本上依據傳統的分類方式能分爲以下六大類型(陳清芳，2001)：

(1) 開機型病毒 (Boot Strap Sector Virus)：

開機型病毒是藏匿和感染磁碟片或硬碟的第一個磁區。因爲DOS的架構設計，使得病毒可以於每次開機時，在作業系統還沒被載入之前就被載入到記憶體中，這個特性使得病毒可以針對DOS的各類中斷(Interrupt)得到完全的控制，並且擁有更大的能力進行傳染與破壞。

(2) 檔案型病毒 (File Infector Virus)：

檔案型病毒通常寄生在可執行檔中(如.com , .exe)。當檔案被執行時，病毒的程式就跟著被執行，而依傳染方式的不同，可分爲常駐型以及非常駐型兩種。

(3) 複合型病毒 (Multi-Partite Virus)：

複合型病毒兼具開機型病毒以及檔案型病毒的特性。它們可以傳染com檔與exe檔，也可以傳染磁碟的開機系統區，其傳染方式是經由傳染的檔案被執行到記憶體的時候，傳染到開機區裡，然後再趁機傳染給其它檔案。

(4) 巨集型病毒 (Macro Virus)：

巨集型病毒主要是利用本身所提供的巨集能力來設計病毒，所以凡是具有寫巨集能力的軟體都具有巨集病毒存在的可能，如Word、Excel等。

(5) 電腦蠕蟲 (Worm)：

電腦蠕蟲也是電腦病毒的其中一種型態，它能自我複製到其它的電腦程式中，並且透過特定的機制經由網路散佈至其它電腦中，與電腦病毒不同的是蠕蟲是透過網路來散佈複製的檔案(Riordan et al.,2005)，而電腦病毒則可透過任何媒介傳染，如磁碟、隨身碟等。蠕蟲是一隻可以獨立運作的程式，具有智慧及自動化的技術，並且結合駭客(Hacker)的手法與電腦病毒的特性，掃描和攻擊網路上的主機(Sihan and Weiping , 2005)。

(6) 特洛伊木馬 (Trojan Horse)：

特洛伊木馬病毒是將惡意編碼嵌進一段程式當中，以執行惡意的活動(Casavant and McMillin,1989)，它不像病毒會感染其它檔案，而是會以一些特殊的管道或是僞裝自己進入使用者電腦中，然後進行伺機執行惡意的行爲或是竊取重要資料等。

近年來病毒的威脅不斷的攀升，根據賽門鐵克防毒軟體公司最新一期的威脅研究報告指出，2007年上半年度(一月至六月) 新型態惡意程式增幅了185個百分比。近年來國內外學者也紛紛投入電腦病毒可能的行爲和特徵加以分析，並試圖找出預防方法，例如提出以知識本體支援病毒行爲偵測及知識管理方法，提供完善的電腦病毒行爲知識更新與查詢，並使用RBF類神經網路進行樣式比對，及模糊派翠網路進行病毒的偵測(許見章，2002)。提出一套病毒風險分析模式，使用分群方法，建立出病毒徵兆本體論，並以點對點網路實作該系統(劉幸文，2004)。提出一個電子郵件病毒過濾器，在用戶收到惡意電子郵件病毒前，偵測此威脅，並對新種病毒加以防範(Shih, 2005)。提出以正規劃概念分析(FCA)建構電腦病毒的知識本體，藉由病毒屬性的查詢尋找病毒解決方案(林建宏，2006)。提出七個使用者需要理解的病毒知識指標，應用於所開發出的ECOVP查詢系統中，讓使用者可處裡病毒事件，包含預防和解毒的方法(Madihah, 2006)。提出以郵件病毒

知識本體論概念與概念間的關係形態轉換成模糊派翠網路結構進行推論，以偵測郵件病毒(姜琇森，2007)。

二、 知識本體 (*Ontology*)

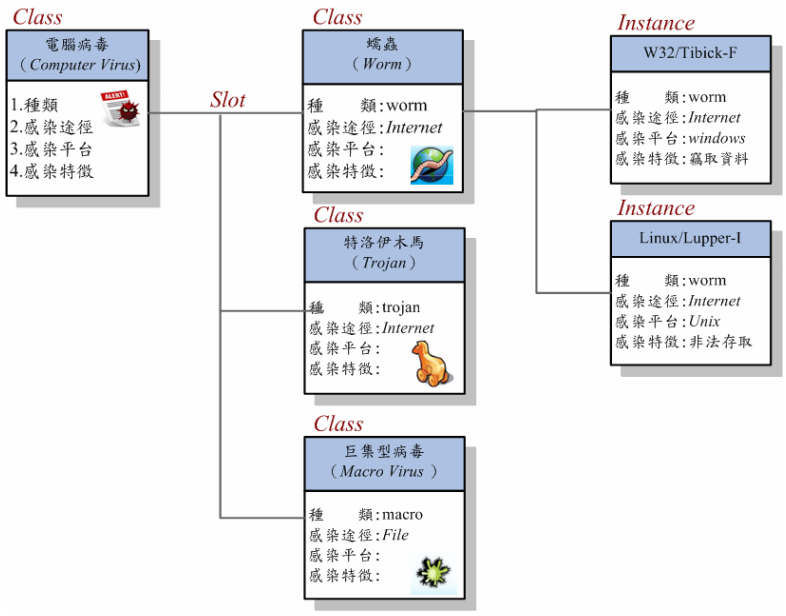
知識本體源自於哲學上探究萬物而加以歸納分析的學說，對萬事萬物的概念加以描述，且說明事物是由哪些物件所組成，以及物件之間彼此的關聯性。學者Guarino (1997)除了認同上述所說的定義，也提出了知識本體是邏輯理論的集合，且是一個明確規格的概念，能夠重複設計及重複使用的知識系統元件。現在有許多學者利用此特性，將此應用在電腦科學領域上，例如知識工程(Knowledge Engineering)、知識表達(Knowledge Representation)、語言工程(Language Engineering)、資訊塑模(Information Modeling)、資訊檢索(Information Retrieval)等(Guarino,1998)。表一爲近年各學者對知識本體所提出的詳細定義。

表一 學者對知識本體的定義

年度	學者	知識本體定義
1993	*Gruber*	知識本體能將概念化的模式詳細的描述，並由術語、定義與相關的公理(Axiom)，組織成分類架構。
1993	*Ushold and Grueninger*	知識本體是一個正式(Formal)且明確的規格，爲大家都能共同接受的概念。
1997	*Swarout*	知識本體是描述一個領域階層結構的概念詞彙(Term)之知識庫框架。
1998	*Guarino*	知識本體是邏輯理論的集合，用以說明字彙(Vocabulary)的特定涵義。

知識本體的建置程序有三項：知識擷取、知識塑模及知識表達(Noy, 1997; Uschold, 1996)，知識擷取(Knowledge Acquisition)方式通常是取自於人類專家或文件書籍的相關知識(Chi, 2007)，並且將知識抽象化的過程，根據應用領域的問題及觀點的不同，也會有不同的擷取方式，目前常見的輔助擷取工具有統計方式、腦力激盪法(Brain storming)、概念構圖(mind mapping)等。知識塑模(Knowledge Modeling)主要將所收集到的知識或資料，建立一個完整的知識網路，使得知識具有階層式的邏輯、關係等架構，目前常見的輔助塑模工具有正規化概念分析(Formal Concept Analysis)、概念圖法(Concept Graph)等。知識表達(Knowledge Representations)是將塑模出來的知識模型轉換成系統能理解的形式呈現在電腦中，常見知識表達方式有規則式(Rule-Base)、框架式(Frame-Based)、物件導向式(Object-Oriented)等。

綜觀來說，知識本體就是清楚描述特定領域內所表達的概念和與概念相關的特徵(Properties)及屬性(Attribute)，並加上屬性的限制(Constraint)和依此分類法所產生的實體(Instance) (Marianne, 1987)。由Noy (2001)等學者認爲，知識本體應有的基本要素爲：Class、Slot、Instance、Axiom。Class 是一個類別或概念，如本研究中的電腦病毒、蠕蟲、特洛伊木馬、巨集型病毒等即可稱之爲類別，其中蠕蟲、特洛伊木馬、巨集型病毒可視爲電腦病毒的子類別，所以電腦病毒可以視爲一個抽象的概念。Slot 在知識本體論中用來描述概念的屬性或概念之間的關聯，如電腦病毒並定會有感染的平台或感染的途徑，或者子類別蠕蟲必定會有很多電腦病毒的種類如Code Red 病毒和Nimda 病毒，而Code Red 病毒和Nimda 病毒屬於蠕蟲的子類別，其中父類別與子類別之間的關聯也可以算是一種Slot。Instance 稱之爲實體或實例，是在知識本體中爲一個概念或類別的案例，實例會繼承類別的所有屬性或關聯，如每一隻蠕蟲都是電腦病毒的實例，但病毒名稱、感染平台及感染途徑都會有所不同。Axiom 於知識本體中是原則或限制，其功能在於制定概念間關聯或限制，與Slot 不同之處在於，Slot 清楚定義兩個類別之間的關聯。圖二爲上述電腦病毒知識本體的基本要素範例圖。

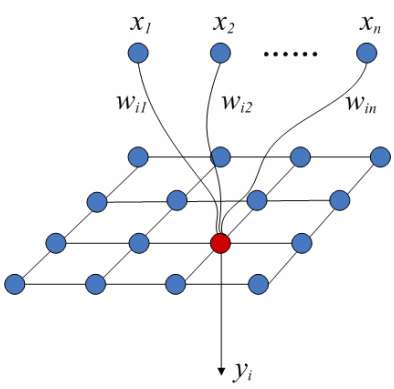


圖二 電腦病毒知識本體基本要素範例圖

三、 機器學習 (*Machine Learning*)

分群主要的目的是用來探索資料中資料聚集關係的一種方法，真實世界中的資料往往都具有群聚的現象，而資料的群聚探索就是希望能挖掘出這個現象。以分群分析而言，輸入的資料常需要轉換爲數字向量，藉由計算、比對向量間的某種距離來決定輸入向量間的分群關係(Hung, 2008)，而常見的分群技術演算法有

K-平均法（K-Means）、階層式分群法(Hierarchical Method)、自我組織特徵映射網路(Self-Organizing Feature Map Network, SOM)等。Kohonen(1984)提出自我組織特徵映射網路，也是根植於競爭式學習的一種網路，其主要目的是希望將任意維度的輸入資料，經由某種映射關係，映射至一維或二維的特徵映射圖上，並且將輸入資料間彼此的拓樸關係保持在特徵映射圖上，如圖三所示。

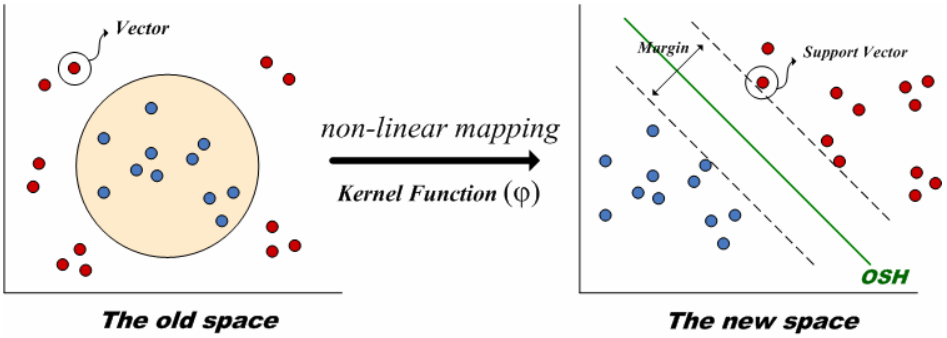


圖三 自我組織特徵映射網路模型

分類是指檢視新資料的特性，然後將其指定到預先定義好的類別中。要進行分類，需將所有類別預先定義好，並有預先分類好的訓練組(Training Set)資料集，分類技術常見的演算法有決策樹(Decision Tree)、類神經網路(Neural Network)、支援向量機(Support Vector Machines, SVM)等。支援向量機是由Vapnik(1999)提出的一個機器學習的法則，其主要的理論是來自統計學習理論中結構風險最小化的原則(Structural Risk Minimization, SRM)。支援向量機的基本概念爲超平面空間(Hyperplane)的切割與核心函數(Kernel Function)的資料型態轉換。

- 超平面空間：假設空間中有兩類資料，若能找到一個平面可將資料分爲二類，則此平面稱之爲超平面空間，如圖四所示。若從中找出一個超平面是能將兩類資料切割後，同時被隔開的最遠的距離，一般稱此平面爲最佳分割超平面(Optimal Separating Hyperplane，OSH)，最佳超平面與兩個分類類別間的距離被稱爲邊界值(Margin)。
- 核心函數：資料爲線性可分割的情況下使用超平面進行分類，但如果需要將非線性的資料加以分類，可採用核心函數來改變資料型態，其主要的概念是將輸入資料由原先的低維度空間(Low dimensional)藉由核心函數(φ)的轉換對映到高維度空間(High dimensional)中，在高維度空間中就可將原本線性不

可分割的資料使用線性可分割的方式一分爲二，換言之，也就是說核心函數(φ) 是將非線性的資料改爲線性資料之後再行分類，如圖四所示。



圖四 非線性分割之轉換

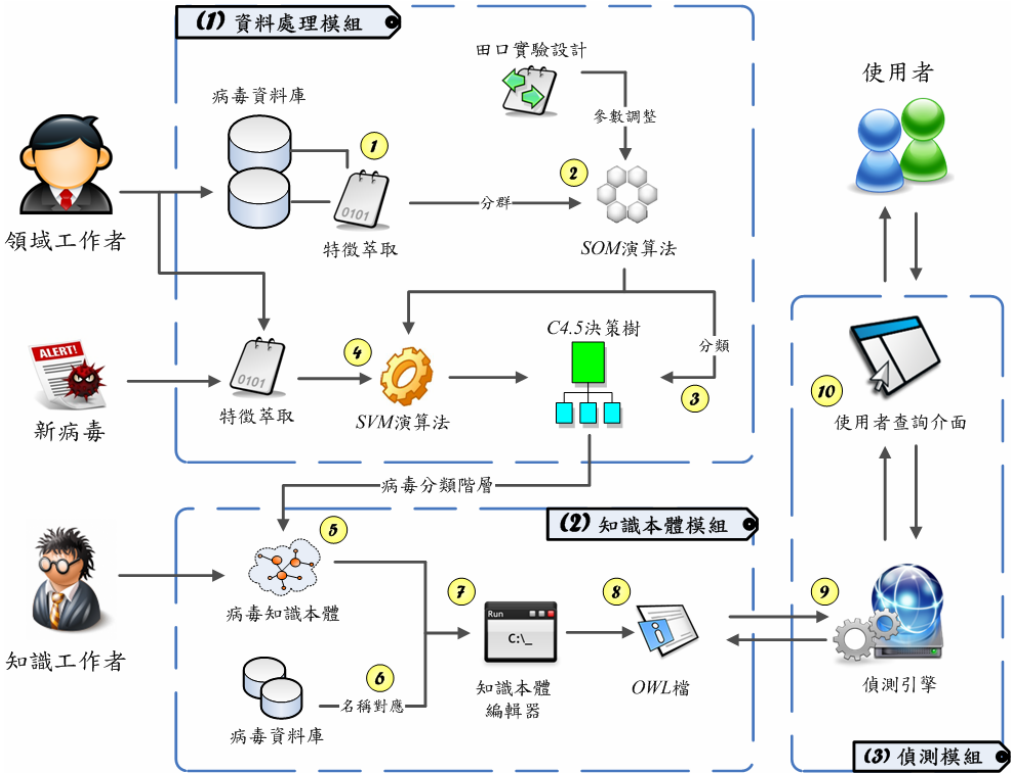
結合SOM與SVM之學習機制是由Cao(2001)學者所提出，應用於時間序列樣式上解決金融方面的問題。其方法是利用"分而制之(Divide and Conquer)"的概念，其中SOM扮演著分割的作用，對於非終端節點的資料，透過二元分割將相似度高的資料做分群，則終端節點若未大於門檻值(Threshold Value)就使用SVM將資料做分類。本研究將收集到的特洛伊木馬病毒資料套用此概念，先對電腦病毒特徵做SOM分群，分群後的資料再使用決策樹產生特洛伊木馬病毒知識分類架構，對於新型的電腦病毒則利用SVM預測電腦病毒隸屬哪個分群的結果。

參、研究設計

本研究利用知識本體技術，建置出一套以專業資訊人員爲導向的電腦病毒知識庫系統。由於目前電腦病毒領域的描述過於專業化，而病毒屬性的描述上也都是由防毒軟體廠商所自行定義，一般使用者較難以理解其描述的意義，因此本系統主要目的是期望能夠滿足專業資訊人員在於電腦病毒分類、偵測及查詢上的需求，而非一般使用者。圖五爲本研究的系統架構圖，其功能描述如下：

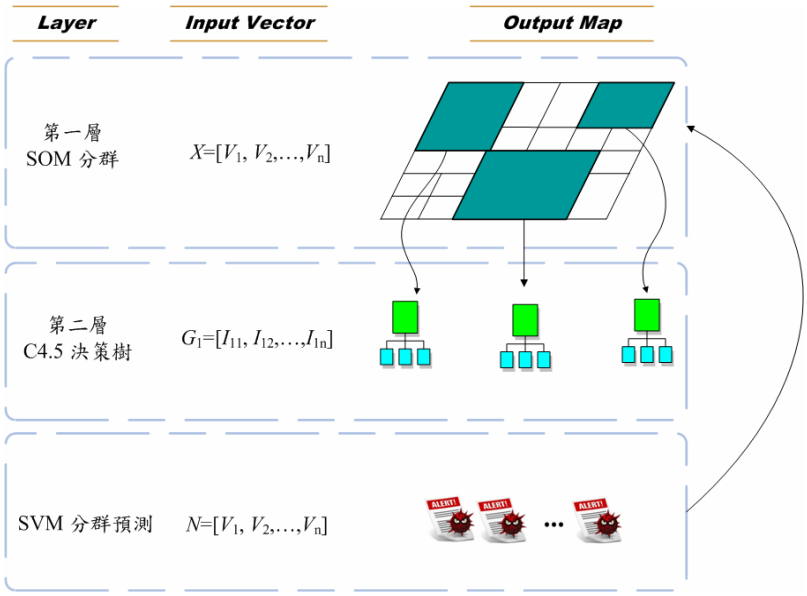
(1) 資料處理模組：首先經由領域工作者收集防毒軟體廠商所提供的病毒資料庫資訊，或經由專家提供的電腦病毒資訊，從中萃取出特洛伊木馬病毒的特徵屬性，萃取出的病毒特徵則轉換成病毒樣式編碼，進行完資料前至處裡後，依此編碼進行SOM病毒分群，分群後再由C4.5 決策樹產生出特洛伊木馬病毒知識分類架構，SVM則進行預測新型態特洛伊木馬的之處裡，最後經由不斷反覆學習的過程，產生出完善的知識分類架構，其流程如圖六所示。本研究的SOM模型取用網路病毒資料庫所萃取出的病毒特徵資料，第一層SOM發展

針對特洛伊木馬病毒分群，並將群聚中擁有最高權重的特徵名稱，作爲SOM輸出單元名稱，第二層或以下的各層SOM發展，則採用C4.5 決策樹分類，其中每個葉節點將爲分類名稱，此模組的研究過程如下：



圖五 系統架構圖

1. 領域工作者收集防毒軟體廠商所提供的病毒資料庫資料X，X=[V_1, V_2,…,V_n]，V_n爲病毒特徵種類，如Windows XP、E-mail感染、傳送檔案等，n爲變數種類總數。每一種病毒特徵種類包含兩個變數，如V=[p_1, p_2]，p_1即爲含有此項特徵，反之p_2即不含有此項特徵。
2. 以X=[V_1, V_2,…,V_n]爲SOM輸入資料，產生第一層SOM分群。
3. 命名所有 SOM 分群名稱，命名方式採取多數決，即每群之中擁有最多此特徵名稱當作分群名稱。
4. 以每群的資料集合作爲C4.5 決策樹輸入資料集，如G_1=[I_{11}, I_{12},…,I_{1n}]，G_1爲SOM分群結果的第一群資料集合，I_{11}爲第一群中的第一筆病毒資料。
5. C4.5 決策樹分類結果，採用每個葉節點名稱爲分類標籤名稱。
6. 新型特洛伊木馬病毒，則採用 SVM 預測落於哪個分群結果。
7. 重複步驟 3 至 5，直到滿足決策者的終止條件爲止。



圖六 資料處理模組流程

(2) 知識本體模組：將資料處裡模組最後產出的特洛伊木馬病毒階層架構，做爲病毒知識本體建置的來源，知識工程師即可參照此架構塑模知識本體，且將其它防毒軟體資料庫中的同種名稱電腦病毒利用知識本體的特性來對應其中之隱含關係，利用知識本體編輯器建置病毒知識本體，最後產生 OWL 檔能被偵測引擎推論。

(3) 偵測模組：提供網頁偵測介面系統給使用者輸入病毒特徵描述之平台，使用者給予特定特洛伊木馬病毒特徵之後，經由偵測引擎推論病毒知識本體資料庫中符合偵測條件之特洛伊木馬病毒，將其推論結果回傳予使用者，並且提供電腦病毒之解決方案。

肆. 系統實作

系統實作將依照本研究所提出的系統架構，依照此流程加以建置病毒知識庫系統。在病毒資料來源部份，本研究希望透過客觀且具公信力的病毒特徵描述，參照目前現有知名防毒軟體廠商(Avira、Sophost、Kaspersky)所提供的特洛伊木馬病毒相關資訊，經由此領域工作者從中萃取出病毒特徵，如表二即是本研究針對所收集到的 200 筆特洛木馬病毒，從中萃取出的 29 個病毒樣式特徵，且將 29 個特徵區分爲感染平台(Platform)、感染途徑(Infects)與感染特徵(Characteristic)三部份。將病毒特徵萃取出之後，即可針對此特徵彙整編列出病毒樣式編碼，如表三所示，即完成輸入資料前至處裡部份。

表二 特洛伊木馬病毒特徵彙總表

感染類別	變數	樣本數	百分比%
感染平台 (Platform)	Windows 98	191	95.5
	Windows XP	194	97
	Windows 2000	193	96. 5
	Windows 2003	172	86
感染途徑 (Infects)	無特殊形式	151	75.5
	E-mail 感染	44	22
	區域網路感染	6	3
	通訊軟體	1	0.5
感染特徵 (Characteristic)	傳送檔案	65	32.5
	傳送惡意檔案	53	26.5
	下載檔案	72	36
	下載惡意檔案	28	14
	關閉防毒軟體	37	18.5
	竊取資訊	75	37.5
	鍵盤側錄	18	9
	變更登錄檔	137	68.5
	中斷應用程式	5	2.5
	降低安全設定	34	17
	啓動郵件引擎	29	14.5
	遠端控制	54	27
	變更存取權限	10	5
	無法連線防毒軟體網站	14	7
	更改 Hosts file	13	6.5
	中斷系統程序	22	11
	開啓通訊埠	90	45
	增加檔案	123	61.5
	自我複製	101	50.5
	刪除檔案	29	14.5
	利用軟體漏洞	3	1.5

表三 部分特洛伊木馬病毒樣式編碼

病毒名稱	病毒樣式編碼
TR/Drop.Small.ako	1,1,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
TR/Dldr.Adload.J.14	1,1,1,1,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
TR/PSW.VB.HU	1,1,1,1,1,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0,0,0,0,1,1,0,0
TR/Spy.Small.DG.8	1,1,1,1,1,0,0,0,0,1,1,0,0,1,1,1,0,0,0,0,0,0,0,0,1,1,0,0,0
TR/Sirery.A	1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,1,0,0
TR/Spy.Goldun.HW	1,1,1,1,1,0,0,0,1,1,0,0,0,1,0,1,0,0,0,1,1,0,0,0,1,1,0,0,0
TR/IRCBot.MX	1,1,1,1,0,1,0,0,0,0,0,0,0,1,0,1,0,0,1,1,0,0,0,0,0,0,1,0,0
TR/Drop.Bagle.FT	1,1,1,1,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0

一、資料處理模組

將病毒樣式編碼做爲 SOM 演算法的輸入值，即爲 SOM 演算法的訓練樣本，透過此網路的訓練學習來進行群集的作用，已得到初次階段分群的結果。本研究實驗軟體採用 Matlab R2006a 中的 SOM 分析工具，實驗樣本將取出特洛伊木馬 200 筆爲實驗數據。爲了使得分群效果有較佳的品質，將採用田口實驗設計法取得一組較佳的參數組合，研究中選取四個對於 SOM 演算法分群品質有影響甚大的參數：拓樸矩陣、鄰近半徑、學習次數及學習率，並在每個網路參數下設定三個水準，如表四所示。

表四 因子水準表

水準	拓樸矩陣	鄰近半徑	學習次數	學習率
水準一	2×5	10	500	0.8
水準二	4×4	20	1000	0.75
水準三	4×5	25	1000	0.7

本研究利用樣本空間座標與所屬群心座標的總距離平均值評估 SOM 網路的分群品質，即爲樣本資料與所屬群心的密集程度，其計算式如下所示：

$$d_j = \frac{\sqrt{\sum_i (X_i - W_j)^2}}{n_j} \tag{1}$$

其中，d_j 爲第 j 群中平均每個樣本點對所屬的群心距離，X_i 爲第 i 個樣本點，W_j 爲第 j 群的群心位置，n_j 爲第 j 群的所有樣本數。然後利用 d_j 求出總距離平均值其公式如下：

$$AD = \frac{\sum_j d_j}{N} \tag{2}$$

則 N 爲分群數，AD 爲總距離平均值即爲分群品質。由於本研究的分群品質爲表示樣本與群心的密集程度的指標，因此當距離越小，表示群內密集程度越高，也代表分群品質越高，因此在田口實驗設計法中，其 SN 比屬於望小特性，其公式如下：

$$SN = -10\log\{\frac{1}{n}\sum_{i=1}^{n} AD_i^2\} \quad (3)$$

其中，*AD*爲總距離平均值，經過九次實驗結果如表五所示。

表五 SOM 實驗結果

實驗	拓樸矩陣	鄰近半徑	學習次數	學習率	分群品質	*SN*
1	10×11	1	500	0.8	4.46643705	-12.99922435
2	10×11	2	850	0.75	3.70310517	-11.37132093
3	10×11	3	1000	0.7	3.39753706	-10.62328405
4	10×10	1	850	0.7	4.24176962	-12.55094155
5	10×10	2	1000	0.8	3.81650618	-11.63331940
6	10×10	3	500	0.75	3.38143519	-10.58202136
7	10×9	1	1000	0.75	4.16021641	-12.38231845
8	10×9	2	500	0.7	3.58902402	-11.09952730
9	10×9	3	850	0.8	3.39913433	-10.62736656

則此實驗所得到的 SN 比回應表與 SN 比回應圖，如表六所示。由 SN 比回應表中可得知最佳的因子參數水準組合爲拓樸矩陣 10×9、鄰近半徑 3、學習次數 850、學習率 0.7。

表六 SN 比回應表

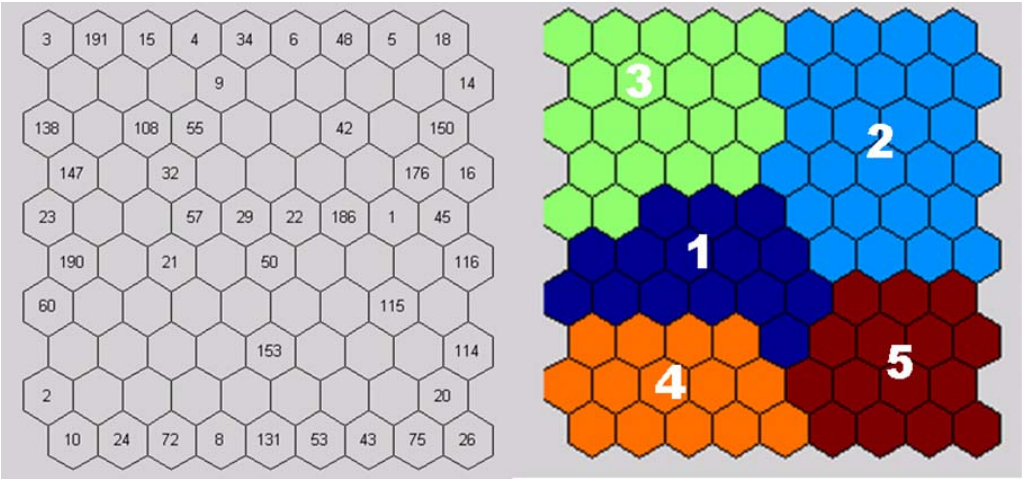
水準因子	拓樸矩陣	鄰近半徑	學習次數	學習率
水準一	-11.66460978	-12.64416145	-11.56025767	-11.75330344
水準二	-11.58876077	-11.36805588	-11.51654301	-11.44522025
水準三	-11.36973744	-10.61089066	-11.54630730	-11.42458430
效應	0.29487234	2.03327079	0.04371466	0.32871914
排名	3	1	4	2

然而要避免實驗中產生交互作用的情況，本研究將參數設計中的各因子所造成效果相近的水準參數挑選出來，再進行一次全實驗，則由於學習次數與學習率所產生的結果跟同水準中的數值相近，因此及挑選這兩組參數進行全實驗，實驗結果如表七所示。

表七 全實驗結果

實驗	拓樸矩陣	鄰近半徑	學習次數	學習率	分群品質
1	10×9	3	500	0.8	3.3667935
2	10×9	3	500	0.75	3.35909745
3	10×9	3	500	0.7	3.40583521
4	10×9	3	850	0.7	3.33128152
5	10×9	3	850	0.8	3.39913433
6	10×9	3	850	0.75	3.40954447
7	10×9	3	1000	0.75	3.59050173
8	**10×9**	**3**	**1000**	**0.7**	**3.29934793**
9	10×9	3	1000	0.8	3.44308644

經由表七全實驗結果，可從中得知實驗 8 的分群品質較接近群心位置，則最佳的參數組合爲拓樸矩陣 10×9、鄰近半徑 3、學習次數 1000、學習率 0.7，此參數組合將 200 筆特洛伊木馬病毒分爲五群，圖七即爲分群結果。



圖七 SOM 分群結果

接著經由 SOM 演算法所得的分群結果作爲分類模型的依據，此步驟的分類演算法是使用 C4.5 決策樹的方式來對病毒特徵進行分類。表八爲特洛伊木馬病毒經由 C4.5 決策樹分類出來結果，其中 SOM 演算法參數的選擇爲：拓樸矩陣 10×9、鄰近半徑 3、學習次數 1000、學習率 0.7。針對新病毒的預測分析，本研究將採用 Vapnik 所提出的 SVM 演算法，針對新型病毒在於分群上的預測。由於現實的空間中不容易將資料做分類，尤其當病毒種類的特徵偏多時此時不容易將資料做線性分割，因此則需要透過核心函數將高維資料轉換至低維度，以利做線性切割，其中本研究採用的核心函數爲多項式核心(Polynomial)函數(Vapnik,

1999)，其公式如下所示：

$$K(x_i, x_j) = (\gamma \cdot x_i \cdot x_j + coef)^d \quad , \gamma > 0 \tag{4}$$

其中 γ 和 coef 爲常數，d 則是多項式的階數。

本研究利用特洛伊木馬病毒 200 筆分群出來的結果，選取 160 筆樣本數當作訓練資料集，40 筆爲測試資料集，使用軟體爲 Weka 3.5.6 版中的 SVM 分類工具，經由實驗結果證實 SVM 演算法對於分群預測結果準確率有 90%，相較於 SOM 演算法與 C4.5 決策樹的預測率都有明顯提昇，表九爲此三種演算法的預測結果。

表八 特洛伊木馬分類結果

分群結果	分類結果	樣本數	分群結果	分類結果	樣本數
1. 啓動郵件引擎	增加檔案型	3	6. Windows 2000	自我複製型	9
				更改 Hosts file 型	2
				啓動郵件引擎型	2
2. 變更登錄檔	Email 型	3		無特殊形式型	6
	Windows 98 型	8		傳送檔案型	12
	下載檔案型	7		竊取資訊型	5
	刪除檔案型	6	5. 自我複製	下載惡意檔案型	2
	遠端控制型	6		自我複製型	14
	傳送惡意檔案型	13		竊取資訊型	2
	傳送檔案型	5		變更登錄檔型	3
	增加檔案型	7			
	鍵盤側錄型	2			
	關閉防毒軟體型	21			
3. 增加檔案	Windows 2003 型	17			
	降低安全設定型	3			
	開啓通訊埠型	9			
	遠端控制型	3			
	傳送惡意檔案型	7			
	增加檔案型	11			
	竊取資訊型	8			
	變更存取權限型	4			

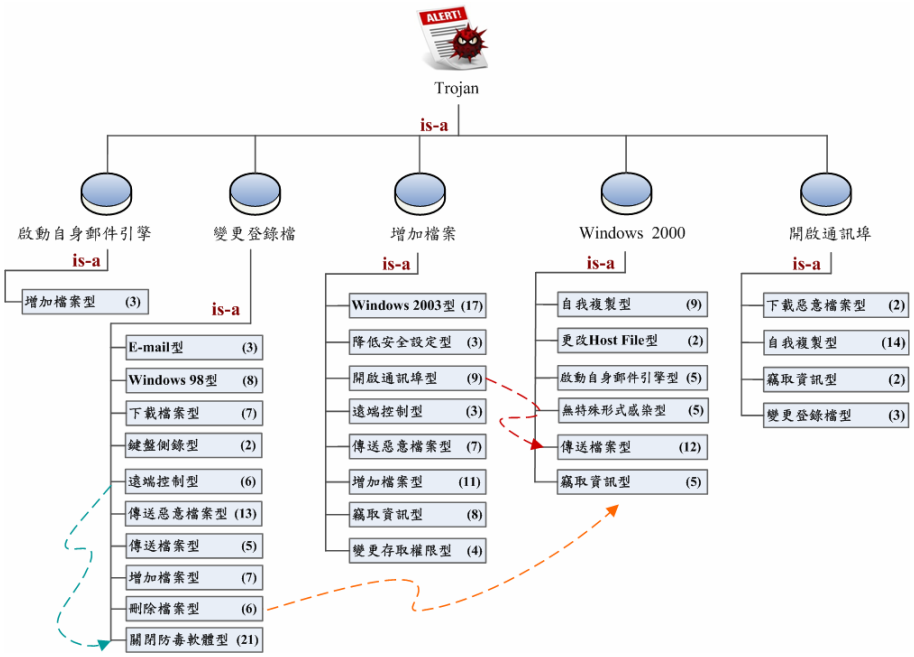
表九 三種演算法預測率比較表

演算法	SOM	C4.5 決策樹	SVM
模型準確率 (訓練集)		90.625%	98.125%
模型預測率 (測試集)	62.5%	75%	90%

二、知識本體模組

本研究知識本體的建置上，經由上一節資料分群分類分析，所提供的病毒分類階層架構加以建置而成，並且透過史丹佛大學發展的知識本體編輯器-Protégé 軟體 3.3.1 版及 OWL 註標語言，作爲開發病毒知識本體的工具。依照特洛伊木馬病毒而言，其根據病毒特徵資料分析結果分成五群類別的特洛伊木馬病毒，其病毒階層如圖八所示，此知識概念圖描述出父類別與子類別間的關係，父類別即爲每個分群出的概念，則子類別即爲分類出來的結果，在依照此架構將每個特洛伊木馬病毒實體建立在分類之中，並在每個實體中描述出此病毒特有的屬性。

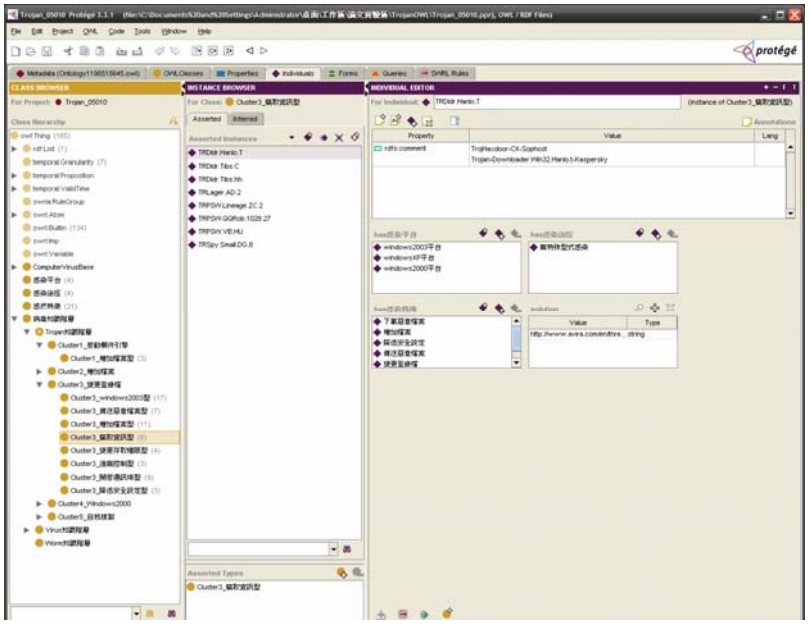
知識本體與一般資訊檢索系統較爲不同的地方，在於知識本體能夠推論隱含關係，以圖八而言，假設有一隻病毒具有「刪除檔案型」的功能，其中「刪除檔案型」則屬於「變更登錄檔」的其中一種分類，而對資訊檢索系統而言，雖可以查詢到此概念，但並不能得知此兩者之間的關係，則從知識本體中即可得知，「刪除檔案型」屬於「變更登錄檔」中的一個分類，此分類都繼承「變更登錄檔」此項功能。



圖八 特洛伊木馬病毒知識概念圖

在圖八中虛線部份，知識本體中可推論出同特徵或同時間點所變種出來的特洛伊木馬病毒，然而在整體分群與分類過程可能歸屬在不同類型下，但透過知識本體的推論將可表示它們之間爲同特徵或同時間點變種出之特洛伊木馬病毒，因此當偵測到此種病毒的特徵時，亦可以使用相同的解決方案去解決此病毒的感染。對於目前各廠商病毒命名不一致問題，在建置每個知識本體的實體時，即可

把各家的名稱描述在病毒的實例中，當使用者發現病毒卻找不到解決方案時，則可透過查詢方式，將此病毒特徵描述在系統中，並找到不同廠商對於同隻病毒描述所提供的解決方案。圖九即爲使用 Protégé 軟體輔助知識工程師所建置出的電腦病毒知識本體，圖中左側即爲各分群分類的概念，而“Cluster3_變更登錄檔”即爲其父類別分群的概念，“Cluster3_竊取資訊型”則爲子類別分類的概念；圖中中間部分即是此類別下的病毒實例，圖中右上則爲註記病毒實體之其它名稱；右下側即爲針對概念所賦予之屬性限制式，屬性特性則有 has 感染平台、has 感染途徑、has 感染特徵與解決方案網址。



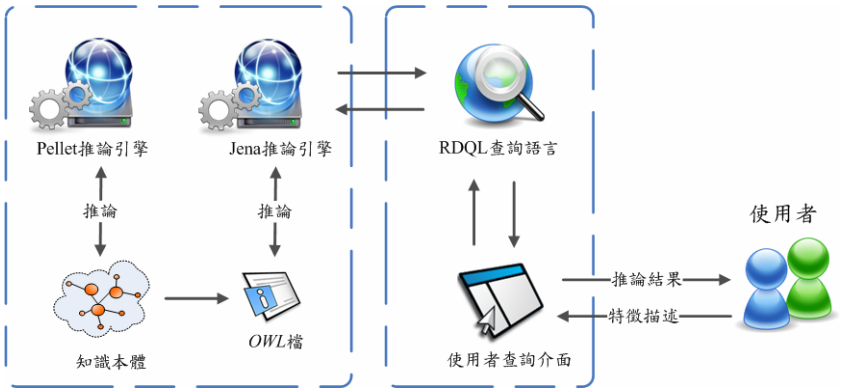
圖九 Protégé 軟體實作電腦病毒知識本體

三、偵測與查詢模組

目前現有的病毒偵測技術中，主要常見技術有病毒碼比對法(Matching virus definition patterns)、完整性檢查法(Check sum)、即時輸出輸入掃描法(Real time I/O scan)、啓發式分析法(Heuristics method)等，而現今的防毒軟體對於病毒的偵測都以病毒碼比對法爲主，此技術是以病毒中一段具有特色或特殊樣式的程式碼，透過防毒軟體中既有的病毒碼資料庫做病毒特徵或樣式比對，當發現異常現象時即可找出感染的病毒。此種技術的優點是能迅速且正確偵測病毒，不會發生誤判的情形，並且只要透過更新病毒碼即可防範新病毒的威脅；缺點則是使用者必須持續更新病毒碼資料庫，若無更新病毒碼則不可防範新型態病毒，且當病毒碼資料庫越龐大時，偵測與搜尋的效率也相對的變差。

本研究將採用知識本體技術的優點，透過知識本體對於未知型態病毒特徵的推論，找到現有病毒知識本體庫中可能的解決方案。知識本體庫與一般資料庫的

不同在於知識本體庫中將每個病毒實體視爲一個概念，每個實體不只是扮演著單獨資料的角色，透過概念化的描述，可將階層上下關係、左右關係表達出來，也因此透過知識本體的技術，資訊人員可在使用者介面上，將所發現的病毒特徵描述在介面中，透過資源描述框架查詢語言(A Query Language for RDF, RDQL)，使用在 Jena 推論引擎去解譯病毒知識本體中的邏輯與規則，並推論出隱含關係，由於 Jena 無法表達與理解描述邏輯的缺點，因此將在搭配 Pellet 推論引擎加強整體的推論效能，所以本研究將採用 Jena 推論引擎與 Pellet 推論引擎的 API，幫助系統能有更多偵測推論的功能，其流程如圖十所示。



圖十 系統偵測流程圖

伍. 結論與建議

本研究提出藉由知識本體及機器學習的方式，進行電腦病毒領域的偵測與查詢，電腦經由既有的病毒特徵，仿造人類學習的模式產生電腦病毒的階層架構，知識工程師只需參照這階層架構即可建置此領域的知識本體，最後利用知識本體的優點，推論出電腦病毒間隱含的關係，並且提供相關的解決方案給予使用者。本研究利用知識本體結合機器學習的方式，建構出電腦病毒知識架構，其優點具有下列幾項：

(1). 有別於傳統防毒軟體的偵測方式，需要透過病毒碼更新，才能適時的偵測及防範最新的電腦病毒，經由知識本體對於既有的電腦病毒特徵做反覆性學習，並且推論出之間的隱含知識，即可偵測出未知型態的電腦病毒。

(2). 經由機器反覆學習的模式，可以迅速且便利的塑模知識架構，知識工程師即可參照客觀的方式，建置特定領域的知識本體。

(3). 對於資料的更新與管理上，經由知識本體的特性，利於知識工程師維護及擴展整體的知識架構。最後本研究不僅解決目前市面上防毒軟體廠商命名不一致的問題，並且提供一個查詢介面，使用者可以經由此介面查詢與偵測電腦病毒。

參考文獻

1. Berners-Lee T., Hendler J. and Lassila O.(2001), “The Semantic Web”, Scientific American, Vol. 284, No. 5, 33-43.

2. Cao L.J.(2003),“Support Vector Machines Experts for Time Series Forecasting” , Neurocomputing, Vol. 51, 321-339.

3. Casavant T.L. and McMillin B.M.(1989), “Safe Computing” , Potentials IEEE, Vol. 8 , No. 3, 29-31.

4. Chi Y.L.(2007),“Elicitation Synergy of Extracting Conceptual Tags and Hierarchies in Textual Document” , Expert Systems with Applications, Vol. 32, No. 2, 349–357.

5. Chi Y.L.(2008),“A Consumer-centric Design Approach to Develop Comprehensive Knowledge-based Systems for Keyword Discovery” , Expert Systems With Applications.

6. Cohen F.(1987), “Computer Viruses Theory and Experiments” , Computer and Security , Vol. 6 , No. 1 , 22-35.

7. Guarino N.(1998), “Formal Ontology and Information Systems” , in Proc.1th , Formal Ontology in Information Systems, Italy, 03-15.

8. Guarino N.(1997),“Understanding Building and Using Ontologies: A Commentary to Using Explicit Ontologies in KBS Development” , International Journal of Human and Computer Studies, Vol. 46, 293-310.

9. Hung C.l. and Tsa, C.F.(2008), “Market Segmentation Based on Hierarchical Self Organizing Map for Markets of Multimedia on Demand” , Expert Systems With Applications, Vol. 34, No. 1, 780-787.

10. Kohonen T.(1984), “Self Organizing and Associative Memory,” Springer-Verlag, Berlin.

11. Madihah S. and Nazean J.(2006), “Knowledge Structure on Virus for User Education”, IEEE Computational Intelligence and Security, Vol. 2 , No. 3-6, 1515-1518.

12. Maedche A. and Staab S.(2001), “Ontology Learning for the Semantic Web ” , IEEE Intelligent Systems, Vol. 16, No. 2, 72-79.

13. Marianne L.(1987), “ The Knowledge Acquisition Grid: A Method for Training Knowledge Engineers”, International Journalof Man-Machine Studies, Vol. 26,

No. 2, 245-255.

14. Noy N.F. and Hafner C.D.(1997), “The State of the Art in Ontologydesign: A Survey and Comparative Review” , AI Magazine, Vol. 18, No. 3, 53–74.

15. Noy N.F. and McGuinness D.L.(2001), “ Ontology Development 101: A Guide to Creating Your First Ontology”, Stanford KS Lab, California, Tech Rep. KSL-01-05.

16. Riordan J., Wespi A. and Zamboni D.(2005),“ How to hook worms (computer network security)” , Spectrum IEEE , Vol. 42 , No. 5, 32-36.

17. Shih D.H., Chiang H.S. and Yen C.D.(2005), “Classification Methods in the Detection of New Malicious Emails,” Information Sciences, Vol. 172, No. 1-2, 241-261.

18. Sihan Q. and Weiping W.(2005), “A survey and trends on internet worm,” Computers and Security, Vol. 24, No. 4, 334-346.

19. Swartout W. and Austin T.(1999), “Ontologies”, IEEE Intelligent Systems, Vol. 14, No. 1, 18-19.

20. Uschold M. and Grueninger M.(1996),“Ontologies: Principles, Methods and Applications ”, Knowledge Engineering Review, Vol. 11, No. 2, 93–155.

21. Vapnik V.N.(1999), “An Overview of Statistical Learning Theory”, IEEE Transactions on Neural Networks, Vol. 10, No. 5, 988-999.

22. 林建宏(2006)，正規劃概念分析建構電腦病毒特徵之知識本體，國立雲林科技大學資訊管理研究所碩士論文。

23. 姜琇森、施東河、黃信銓(2007)，以本體論爲基礎之惡意郵件偵測，資訊管理學報，第十四卷，專刊，頁01-28。

24. 戚玉樑(2006)，以本體爲核心的圖像註記應用於知識化資訊檢索，國科會專題研究報告，(NSC95-2416-H-033-009)，桃園。

25. 許見章、李中彥、文琪平(2002)，以本體論支援電腦病毒行爲偵測及其知識管理， 國科會專題研究報告，(NSC91-2623-7-034-002)，台北。

26. 陳清芳(2002)，電腦病毒紅皮書，臺北，趨勢教育。

27. 劉幸文(2004)，以病毒徵兆本體論輔助電腦病毒風險分析，輔仁大學資訊管理研究所碩士論文。