Available online at www.sciencedirect.com
ScienceDirect
Data & Knowledge Engineering 60 (2007) 547–566
DATA & KNOWLEDGE ENGINEERING
www.elsevier.com/locate/datak

Automated ontology construction for unstructured text documents

Chang-Shing Lee [a,*], Yuan-Fang Kao [b], Yau-Hwang Kuo [b], Mei-Hui Wang [a]

[a] *Department of Computer Science and Information Engineering, National University of Tainan, 33, Sec. 2, Shu-Lin St., Tainan 700, Taiwan*
[b] *CREDIT Research Center, National Cheng Kung University, Tainan, Taiwan*

Received 1 April 2006; accepted 2 April 2006
Available online 2 May 2006

Abstract

Ontology is playing an increasingly important role in knowledge management and the Semantic Web. This study presents a novel episode-based ontology construction mechanism to extract domain ontology from unstructured text documents. Additionally, fuzzy numbers for *conceptual similarity* computing are presented for concept clustering and taxonomic relation definitions. Moreover, concept attributes and operations can be extracted from episodes to construct a domain ontology, while non-taxonomic relations can be generated from episodes. The fuzzy inference mechanism is also applied to obtain new instances for ontology learning. Experimental results show that the proposed approach can effectively construct a Chinese domain ontology from unstructured text documents.

Keywords: Ontology construction; Ontology learning; Concept clustering; Fuzzy inference; Chinese natural language processing

1. Introduction

Ontology is an essential part of many applications. Supported by an ontology, both the user and the system can communicate with each other using a common understanding of a domain [4,19]. Although ontology has been proposed as an important and natural means of representing real-world knowledge for the development of database designs, most ontology constructions are not performed either systematically or automatically [17]. Information systems increasingly depend on ontology to structure data in a machine-readable format and ensure satisfactory performance. Some generic ontologies, like WordNet [13] and Cyc [9], are available, but most applications need a specific domain ontology to describe concepts and relations in that domain. Automatic ontology construction is a difficult task owing to the lack of a structured knowledge base or domain thesaurus. Ontology construction traditionally depends on domain experts, but it is lengthy, costly and controversial [15]. While many ontology tools, like OntoEdit [18], Protege-2000 [16] and Ontolingua [2], are available to aid the construction of ontologies, ontology construction still needs human effort. Most

* Corresponding author. Tel.: +886 6 2606123x7709; fax: +886 6 2606125.
E-mail addresses: leecs@mail.nutn.edu.tw, leecs@cad.csie.ncku.edu.tw (C.-S. Lee).

0169-023X/$ - see front matter
doi:10.1016/j.datak.2006.04.001

studies of ontology construction and application assume manual construction, and only a few have proposed automatic methods [5].

Various ontology construction approaches have been presented recently. For instance, Khan and Luo [5] constructed an ontology using a modified Self-Organization Map (*SOM*) clustering algorithm in a bottom-up fashion. Widyantoro and Yen [20] presented a fuzzy ontology based on fuzzy narrower term relationships and a fuzzy broader term relation for query refinement in an abstract search engine. Yoshinaga et al. [22] automatically constructed an ontology including keywords and relations. Zhou et al. [23] proposed a customizable collaborative system to construct the domain ontology. Maedche and Staab [11] presented an ontology learning framework encompassing ontology import, extraction, pruning, refinement and evaluation. MissiKoff et al. [14] proposed an integrated approach to web ontology learning and engineering, which can construct and access the domain ontology to integrate information intelligently within a virtual user community. Navigli et al. [15] utilized the WordNet and SemCor to interpret complex terms semantically, which it can save the problem of the semantic disambiguation. OntoSeek [31] combined an ontology-driven content-matching mechanism with moderately expressive representation formalism. Lammari and Metais [29] presented a set of algorithms to construct and maintain ontologies. Andreasen et al. [27] described a method and a system for content-based querying of texts based on the ontology. Elliman et al. [34] proposed a method for constructing the ontology to represent a set of web pages on a specified site, using the *SOM* to construct the hierarchy. Hotho et al. [35] proposed various clustering techniques to view text documents with the help of an ontology.

Lee et al. [26] presented a meeting scheduling system based on the personal ontology and the fuzzy meeting scheduling ontology. Furthermore, Lee et al. also presented some approaches for Chinese text processing, for example, an ontology-based fuzzy event extraction agent for Chinese news summarization [8,30], a fuzzy ontology for applying to text summarization [28], and a Chinese term clustering mechanism for generating semantic concepts for a news ontology [25]. The most recent development in standard ontology languages is OWL from the World Wide Web Consortium (W3C). Like Protege-OWL plugin, it not only allows concepts to be described, but also provides new facilities. In addition, OWL has a richer set of operators, including *and*, *or* and *negation*, than other standard ontology languages. Therefore, complex concepts can be built in definitions from simpler concepts. Furthermore, the logical model allows the use of a reasoner to check whether the statements and definitions in the ontology are mutually consistent, and to recognize which concepts fit under which definitions. The reasoner can therefore help maintain the hierarchy correctly. This feature is particularly helpful when handling cases of classes with more than one parent. OWL ontologies may be categorized into three species or sub-languages, namely OWL-Lite, OWL-DL and OWL-Full. A defining feature of each sub-language is its expressiveness. OWL-Lite is the least expressive sub-language, while OWL-Full is the most expressive, with OWL-DL between the other two sub-languages. The expressiveness of OWL-DL falls between that of OWL-Lite and OWL-Full. OWL-DL may be considered as an extension of OWL-Lite and OWL-Full an extension of OWL-DL [36].

This study presents an episode-based fuzzy inference mechanism to extract the domain ontology from unstructured text documents. The proposed approach is an original synthesis of previously reported approaches, including *the concept of the episode*, *the concept clustering for Chinese text documents*, and *the fuzzy inference mechanism*. No effective and efficient approaches to fully automated ontology construction from unstructured Chinese text documents have yet been found. Additionally, the challenge of solving the conceptual meaning of the Chinese term in a sentence is that a Chinese term may comprise many words and that a combination of words in a Chinese term may have different meanings. Take the Chinese terms "行政院新聞局 (Government Information Office of Executive Yuan)" and "電腦科學 (computer science)" for instance. The Chinese term "行政院新聞局 (Government Information Office of Executive Yuan)" has six words, namely "行 (walk)", "政 (policy)", "院 (yuan)", "新 (new)", "聞 (smell)", and "局 (bureau)", and can be split in various ways, such as {"行", "政", "院", "新", "聞", "局"}, {"行政 (administrator)", "院 (yuan)", "新聞 (news)", "局 (bureau)"}, and {"行政院 (Executive Yuan)", "新聞局 (Government Information Office)"}. The Chinese term "電腦科學 (computer science)" has four words, namely "電 (electricity)", "腦 (brain)", "科 (family)" and "學 (learn)", and can be split into {"電", "腦", "科", "學"}, {"電腦 (computer)", "科學 (science)"}, {"電 (electricity)", "腦科 (department of the brain)", "學 (learn)"}, and {"電腦科 (department of the computer)", "學 (learn)"}. However, challenge of processing unstructured text documents is to extract the desired Chinese terms, such as "行政院新聞局 (Government Information Office of Executive Yuan)" or

"電腦科學 (computer science)", from a sentence. Therefore, natural language processing in the Chinese language is very different from that in English. The fuzzy numbers for the *conceptual similarity* computing are presented for concept clustering and defining taxonomic relationships. Moreover, the attributes and operations of concepts can be extracted from episodes for the ontology construction. Non-taxonomic relationships are generated based on episodes. The fuzzy inference mechanism is further adopted to obtain new ontology learning instances. Experimental results indicate that the proposed approach can effectively construct the Chinese domain ontology from unstructured text documents.

This study is organized as follows. Section 2 presents the episode-based ontology construction mechanism. Section 3 proposes the fuzzy inference mechanism for Chinese text ontology learning. The experimental results are shown in Section 4. Finally, conclusions are drawn in Section 5.

2. Episode-based ontology construction mechanism

This section describes the episode-based ontology construction mechanism. Section 2.1 briefly introduces the concept of the episode. The domain ontology is defined in Section 1. Section 2.3 describes the automatic construction process for Chinese domain ontology from unstructured text documents.

2.1. The concept of the episode

The concept of the episode was proposed by Ahonen et al. [1] and Mannila et al. [12]. An episode $\boldsymbol{e}$ is formally defined as a triple $(V, \leqslant, g)$, where V denotes a set of nodes; $\leqslant$ denotes a partial order on V, and $g: V \rightarrow E$ denotes a mapping that associates each node with an event type E. The interpretation of an episode is that the events in $g(V)$ have to occur in the order described by $\leqslant$. An episode $\boldsymbol{e}$ is parallel if the partial order $\leqslant$ is a trivial order (i.e., x *not* $\leqslant y$ for all $x, y \in V$ such that $x \neq y$). Conversely, an episode $\boldsymbol{e}$ is serial if the partial order $\leqslant$ is a total order (i.e., $x \leqslant y$ or $y \leqslant x$ for all $x, y \in V$).

Informally, an episode is a partially ordered collection of events occurring together. Episodes can be directed acyclic graphs. For instance, consider episodes α, β, and γ in Fig. 1. Episode α is a serial episode, and can only occur in a sequence that includes events of type a and b in that order. Episode β is a parallel episode that does not impose constraints on the order of d and e. Episode γ is a non-serial and non-parallel episode, and can occur in a sequence in which occurrences of c precede occurrences of d and e, which may occur in any order [12].

2.2. The definition of domain ontology

This section briefly describes the graph-based definition of the domain ontology. Fig. 2 illustrates the structure of the graph-based domain ontology.

Definition 1 (*Domain Ontology* [28]). A domain ontology defines a set of representational terms called concepts. Inter-relationships among these concepts describe a target world [29]. A domain ontology has four layers, called the *domain layer*, *category layer*, *class layer* and *instance layer* [30]. The *domain layer* denotes the domain name of an ontology, and comprises various categories defined by domain experts. The category layer has many categories, termed category 1, category 2, ..., and category k. Each concept in the *class layer* contains a concept name C_i, an attribute set $\{A_{C_i1}, \ldots, A_{C_iq_i}\}$ and an operation set $\{O_{C_i1}, \ldots, O_{C_iq_i}\}$ for an application domain. In the *instance layer*, each concept contains a concept name C_i, an attribute set

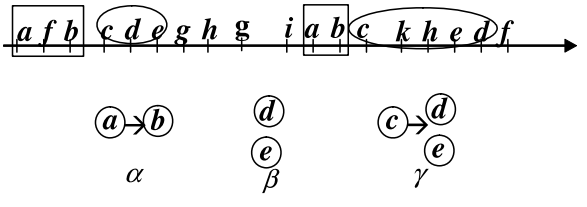


Fig. 1. An example of an episode.

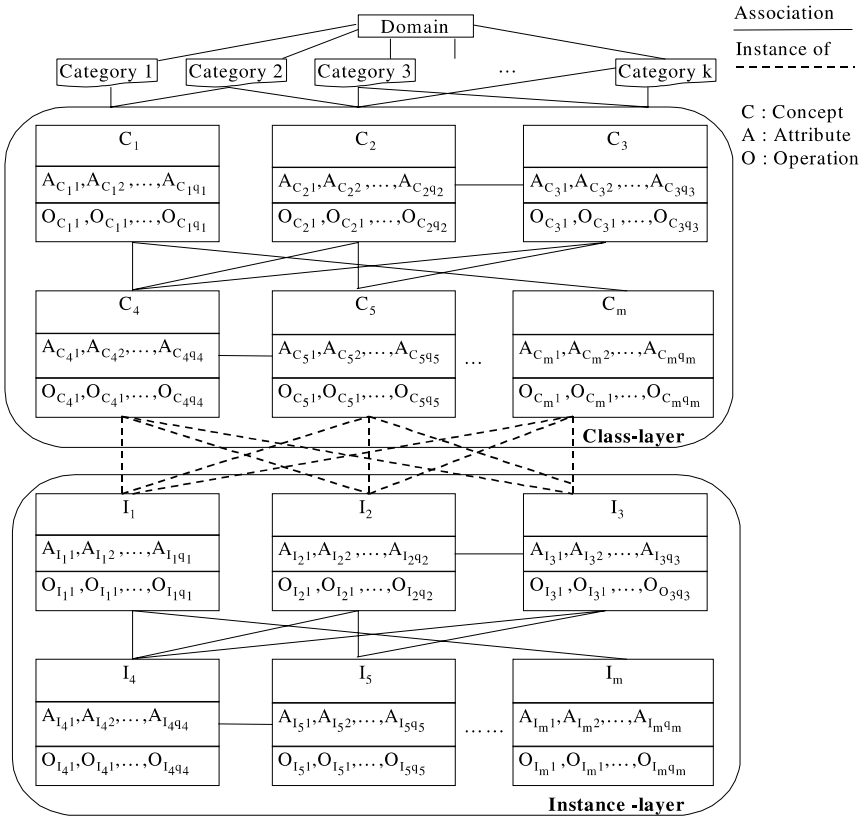


Fig. 2. The domain ontology architecture.

$\{A_{I_i}1, \ldots, A_{I_i q_i}\}$ and an operation set $\{O_{I_i}1, \ldots, O_{I_i q_i}\}$ for an application domain. The domain ontology has two relationships, namely *association* and *instance of*.

2.3. Automatic construction process for domain ontology

This section applies the concept of the episodes to assist in the construction of Chinese domain ontology from unstructured text documents. Additionally, the *SOM* algorithm [5,32] is adopted to cluster the concepts of Chinese terms. Fig. 3 displays the flowchart of the episode-based Chinese domain ontology construction, which includes four processes, namely *Document Pre-processing*, *Concept Clustering*, *Episode Extraction*, and *Attributes-Operations-Associations Extraction*, which are described below.

A. Document pre-processing

The *CKIP* [24] utilized in this study includes a Chinese *Part-of-Speech* (*POS*) *Tagger* and a Chinese news corpus developed by the *CKIP* (Chinese Knowledge Information Processing) Group, a research team in Taiwan formed by the Institute of Information Science and the Institute of Linguistics of Academia Sinica in 1986. The aim of the *CKIP* is to create a fundamental research environment for Chinese natural language processing (http://ckip.iis.sinica.edu.tw/CKIP/engversion/index.htm). The corpus and dictionary provide adequate Chinese *POS* knowledge to analyze the features of the terms for semantic concept clustering. The *Stop Word Filter* is used to reduce significantly the number of Chinese terms, while preserving the terms with partial noun tags or verb tags and filtering the terms with other *POS* tags. The preserved terms used in this study are Na (common noun), Nb (proper noun), Nc (location noun), Nd (time noun) and various classes of verbs (VA, VB, VC, VD, VE, VF, VG, VH, VI, VJ, VK, VL). The filtered terms are Ne (stable noun), Nf (quantity noun), Ng (direction noun), Nh (pronoun), adjective, adverb, preposition, conjunction, particle, and interjection. However, whether a term is preserved or filtered depends on the domain and applications.

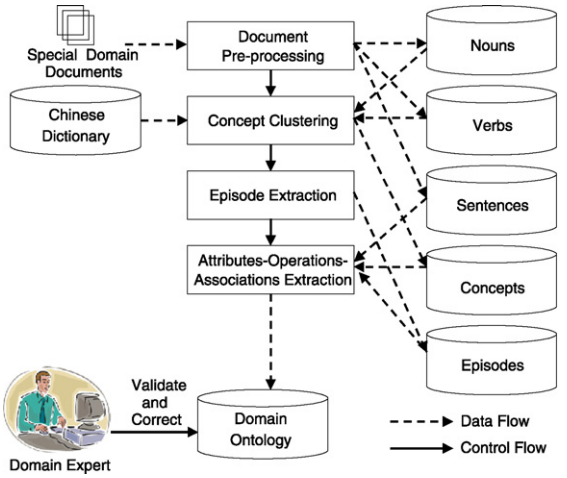


Fig. 3. Flowchart of episode-based Chinese domain ontology construction.

B. Concept clustering

This process aims to cluster concepts and instances from documents. To select important terms for *Concept Clustering*, the nouns with the highest $tf \times idf$ values are preserved and adopted in *Term Analysis*, where *tf* is the term frequency and *idf* is inverse document frequency [3]. In *Term Analysis* process of this study, the three factors, namely *POS*, *Term-Vocabulary* (*TV*), and *Term-Concept* (*TC*) were selected as the conceptual similarity factors for analyzing the Chinese terms and calculating the conceptual similarity between any two Chinese terms based on the features of the Chinese language and the definitions of the *CKIP*. These terms are briefly described as follows. (1) *POS*: Each node of the tagging tree denotes a Chinese *POS* tag defined by *CKIP*. The path length between two nodes is adopted to calculate the conceptual similarity in *POS* between any two Chinese terms. (2) *TV*: The value of the conceptual similarity in *TV* between any two Chinese terms is calculated, according to the characteristics of Chinese language. (3) *TC*: The conceptual structure provided by *CKIP* is adopted to obtain the concept of nouns and calculate the conceptual similarity in *TC* between any two Chinese terms. These factors are described below. Other relevant factors for analyzing Chinese terms are left for future study.

B.1. The conceptual similarity in pos between any two terms. Each node of the tagging tree in Fig. 4 denotes a Chinese *POS* tag defined by *CKIP*. The path length between two nodes is used to calculate the conceptual similarity w_{pos} in *POS* between any two Chinese terms. The path length for any two nodes in the tagging tree is bounded in the interval $[0, 6]$. The value of w_{pos} is large when the path distance of any two Chinese terms is short. For instance, the two terms "電腦 (computer)" and "軟體 (software)" have *POS* values of "Nab" and "Nac", respectively, hence the path distance between them is 2 (Nab → Na → Nac). The conceptual similarity w_{pos} is calculated as follows.

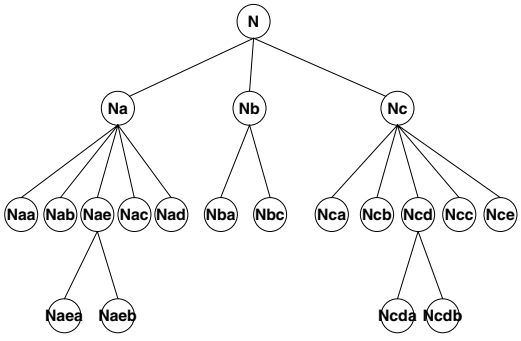


Fig. 4. *POS* tagging tree of *CKIP*.

Calculating the conceptual similarity in *POS* algorithm
Input: All terms $(t_1, t_2, \ldots, t_n)$ selected from $TF \times IDF$ *Selection*
Output: Conceptual similarity w_{pos} in *POS* between any two Chinese terms
Method:
Step 1: Build a *CKIP* tagging tree
Step 2: For all terms $(t_1, t_2, \ldots, t_n)$
  Step 2.1: For all terms $(t_1, t_2, \ldots, t_n)$
    Step 2.1.1: Generate a term pair (t_a, t_b) $1 \leqslant a < b \leqslant n$
    Step 2.1.2: $path =$ length of path between two *POS* tags of (t_a, t_b) in *CKIP* tagging tree
    Step 2.1.3: $w_{\text{pos}} = \frac{LB - path}{LB}$ /* Normalize w_{pos}, where LB denotes the maximum path length in conceptual structure, here $LB = 6$ */
Step 3: End.

B.2. The conceptual similarity in term-vocabulary between any two terms. Almost every word in the Chinese language is a morpheme with its own meaning. This study considers three characteristics of Chinese terms to analyze the conceptual similarity in term-vocabulary between any two terms based on the definition in the *CKIP* [25,28,30]. The three characteristics are: (1) the more identical words in both terms in the pair, the more similar the terms are to each other in semantic meaning; (2) terms in a pair with both identical and continuous words have much greater semantic similarity than those in a pair without identical or continuous words, and (3) terms in a pair with identical starting or ending words have a strong semantic similarity. The conceptual similarity value w_{TV} in TV between any pair of Chinese terms is calculated according to these three characteristics of Chinese language. The algorithm for calculating the conceptual similarity w_{TV} is given below.

Computing the conceptual similarity in term-vocabulary algorithm
Input: All terms $(t_1, t_2, \ldots, t_n)$ selected from $TF \times IDF$ *Selection*
Output: Conceptual similarity w_{TV} in TV
Method:
Step 1: For all terms $(t_1, t_2, \ldots, t_n)$
  Step 1.1: For all terms $(t_1, t_2, \ldots, t_n)$
    Step 1.1.1: Generate a term pair (t_a, t_b) $1 \leqslant a < b \leqslant n$
    Step 1.1.2: $n =$ identical and continuous words between the Chinese term pair (t_a, t_b)
    Step 1.1.3: If $n \geqslant 2$ then $w_{TV} = 1 \times 2^{n-1}$.
    Step 1.1.4: If two Chinese terms have the same starting word then $w_{TV} = w_{TV} + 0.5$.
    Step 1.1.5: If two Chinese terms have the same ending word then $w_{TV} = w_{TV} + 0.5$.
Step 2: $Max_{TV} =$ maximum w_{TV} value of all term pairs.
Step 3: $Min_{TV} =$ minimum w_{TV} value of all term pairs.
Step 4: $w_{TV} = \frac{w_{TV} - Min_{TV}}{Max_{TV} - Min_{TV}}$ /* Normalize w_{TV} */
Step 5: End.

B.3. The conceptual similarity in term-concept between any two terms. The conceptual structure provided by the *CKIP* for Chinese terms is like the hierarchical tree shown in the Appendix [24]. The conceptual structure is adopted to obtain the concept of nouns and calculate the conceptual similarity w_{TC} in TC between any two Chinese terms. The path length for any two nodes in the conceptual structure is bounded in the interval $[0, 12]$. The algorithm for calculating the conceptual similarity w_{TC} is now described as follows.

Computing the conceptual similarity in term-concept algorithm
Input: The conceptual structure, Chinese Dictionary, and all terms $(t_1, t_2, \ldots, t_n)$ selected from $TF \times IDF$ *Selection*
Output: Conceptual similarity w_{TC} in TC
Step 1: Build the conceptual structure
Step 2: For all terms $(t_1, t_2, \ldots, t_n)$

Step 2.1: For all terms $(t_1, t_2, \ldots, t_n)$
Step 2.1.1: Generate a term pair (t_a, t_b) $1 \leqslant a < b \leqslant n$
Step 2.1.2: Obtain the concepts of the Chinese term pair (t_a, t_b) from Chinese Dictionary
Step 2.1.3: $path$ = length of path between two concepts in the conceptual structure
Step 2.1.4: $w_{TC} = \frac{LB-path}{LB}$ /* Normalize w_{TC}, where LB denotes the maximum path length in the conceptual structure, here $LB = 12$ */
Step 3: End.

Next, a well-known clustering method, the *SOM* [6,37], is adopted for concept clustering. An *SOM* is an unsupervised neural network, which maps high-dimensional input data onto a two-dimensional output topological space. Moreover, the *SOM* can be regarded as a "nonlinear projection" of the high-dimensional input data vector onto the two-dimensional display, making *SOM* optimally suitable for visualizing and clustering complex data [5]. Self-organizing neural network-based clustering algorithms have been widely studied for the last decade, but like hierarchical clustering algorithms, have no predefined number of clusters in self-organizing neural networks. The clusters can be distinguished by visualization with manual help. The network size is always greater than the optimal number of clusters in the underlying dataset [32]. A number of self-organizing neural network-based hierarchical clustering algorithms have been presented. The neural network mechanism makes these algorithms robust in terms of noisy data. Additionally, these algorithms inherit the advantages of the original self-organizing map, which easily visualizes the clustering result [32]. In *SOM Clustering*, each input term is expressed as a vector $\{d_{i1}, d_{i2}, \ldots, d_{in}\}$, where d_{ij} denotes the conceptual similarity between any two Chinese terms t_i and t_j, and d_{ij} is bounded in the interval $[0, 1]$. The conceptual similarity values w_{pos}, w_{TV}, and w_{TC} in *POS*, *TV*, and *TC*, respectively, can be obtained according to the *Term Analysis* discussion. Hence, the d_{ij} is calculated using Eq. (1):

$$d_{ij} = w_1 \cdot w_{\text{pos}} + w_2 \cdot w_{TV} + w_3 \cdot w_{TC} \tag{1}$$

Eq. (1) is used to compute the conceptual similarity between two Chinese terms. This study sets $w_1 = 0.3$, $w_2 = 0.3$ and $w_3 = 0.4$, because the authors regard *TC* as more important than *POS* and *TV*. An input term vector $\{d_{i1}, d_{i2}, \ldots, d_{in}\}$ can be mapped to input neurons. Terms with a certain degree of conceptual similarity in word meaning are gathered in the neighboring output neurons. Take Chinese terms "車子(car)", "計程車(taxi)", "公車(bus)", and "火車(train)" for example. All of them have the same Chinese word "車". Through the domain expert and term analysis process of three conceptual clustering factors, *POS*, *TV*, and *TC*, "車子(car)", "計程車(taxi)", "公車(bus)", and "火車(train)" are determined to be subclass concepts of "車輛(vehicle)".

C. Episode extraction

The *Document Pre-processing* process separates the text into sentences, including nouns and verbs, which are then fed into the *Episode Extraction* process to obtain the episodes. This study denotes a term as a triple (*term*, *POS*, *index*), where *index* is the position of this term in the sentence. An episode is extracted if the episode occurs within an interval of a given window size, and the episode's occurring frequency of the text document set is larger than the defined minimal occurrence value. To increase the accuracy of the episodes, the punctuation is filtered and the *POS* of terms with Na, Nb, Nc, Nd and verbs are retained in the sentence. An example of a sentence in a Chinese news document is shown below.

"德國門將卡恩贏得本屆世足賽代表最佳球員的金球獎。"

"Germany keeper Oliver Kahn took home the Golden Ball for best player at the 2002 World Cup."

By the *Document Pre-processing* with the *CKIP* process, the sentence with the terms and *POS* is created as follows:

德國(Nc) 門將(Na) 卡恩(Nb) 贏得(VJ) 本(Nes) 屆(Nf) 世足賽(Nb) 代表(Na) 最佳(A) 球員(Na) 的(DE) 金球獎(Nb)。(PERIODCATEGORY)

By the *Stop Word Filter* process, the terms with triple (*term*, *POS*, *index*) representation are shown below:

(德國, Nc, 1) (門將, Na, 2) (卡恩, Nb, 3) (贏得, VJ, 4) (世足賽, Nb, 5) (代表, Na, 6) (球員, Na, 7) (金球獎, Nb, 8)

Finally, the *episode extraction* process generates the episodes with window size 6 as follows:

德國(Nc)_門將(Na)_卡恩(Nb)
Germany_keeper_Oliver Kahn

卡恩(Nb)_贏得(VJ)_金球獎(Nb)
Oliver Kahn_took_Golden Ball

Yen et al. [21] presented an algorithm to mine the sequential patterns to discover knowledge from large databases. This study extends the algorithm to extract term episodes from Chinese news documents. The episode extraction algorithm is stopped when large 3-sequences are found.

The notation for the episode extraction algorithm is given below:

$\Im < t_1, t_2, \ldots, t_k >$: This set stores the term sequence $t_1, t_2, \ldots, t_k$ occurring in a given sentence.
$\Im < t_1, t_2, \ldots, t_k > .cardinality$: This variable denotes the number of item in $\Im < t_1, t_2, \ldots, t_k >$, and the number of occurrences of the term sequence $t_1, t_2, \ldots, t_k$.
$t_i.position$: denotes the position of t_i in a sentence.
sentence_num: The sequence number of a sentence.

Episode Extraction Algorithm

/* Extract episodes that appear within the given window size with occurrence frequencies above given minimum occurrence from sentences */

Input: Sentences, window size *Window_Size* and minimum occurrence *Minimum_Occurence*
Output: Episodes
Method:
Step 1: Generate *Large 1-Sequence*
  Step 1.1: For all terms t_i
    Step 1.1.1: Scan all sentences
    Step 1.1.2: If t_i appears in this sentence
      Step 1.1.2.1: Record *sentence_num* in $\Im\langle t_i\rangle$
  Step 1.2: If $\Im\langle t_i\rangle.cardinality \geqslant Minimum\_Support$
    Step 1.2.1: Add $\langle t_i\rangle$ to *Large 1-Sequence*
Step 2: Generate *Large 2-Sequence*
  Step 2.1: For all permutations $\langle t_a, t_b\rangle$ where $\langle t_a\rangle$ and $\langle t_b\rangle$ are selected from *Large 1-Sequence*
  Step 2.2: For all sentences with both $\langle t_a\rangle$ and $\langle t_b\rangle$
    Step 2.2.1: If ($t_b.position > t_a.position$ and $t_b.position\text{-}t_a.position \leqslant Window\_Size$)
      Step 2.2.1.1: Record *sentence_num* in $\Im\langle t_a, t_b\rangle$
    Step 2.2.2: If ($\Im\langle t_a, t_b\rangle.cardinality \geqslant Minimum\_Support$)
      Step 2.2.2.1: Add $\langle t_a, t_b\rangle$ to *Large 2-Sequence*
Step 3: *Large k-Sequence* = *Large 2-Sequence*
Step 4: Do {
  Step 4.1: For all $\langle t_1, t_2, \ldots, t_k\rangle$ in *Large k-Sequence* and $\langle t_a, t_b\rangle$ in *Large 2-Sequence*
    Step 4.1.1: If ($t_k = t_a$)
      Step 4.1.1.1: $\langle t_1, t_2, \ldots, t_k, t_b\rangle$ is a candidate of *Large k + 1-Sequence*
      Step 4.1.1.2: For all sentences with both $\langle t_1, t_2, \ldots, t_k\rangle$ and $\langle t_a, t_b\rangle$
        Step 4.1.1.2.1: If ($t_b.position > t_1.position$ and $t_b.position\text{-}t_1.position \leqslant Window\_Size$)
          Step 4.1.1.2.1.1: Record *sentence_num* in $\Im\langle t_1, t_2, \ldots, t_k, t_b\rangle$
        Step 4.1.1.2.2: If($\Im\langle t_1, t_2, \ldots, t_k, t_b\rangle.cardinality \geqslant Minimum\_Support$)
          Step 4.1.1.2.2.1: Add $\langle t_1, t_2, \ldots, t_k, t_b\rangle$ to *Large k + 1-Sequence*
  }while(*Large k + 1-Sequence* can be generated) go to Step 4.1
Step 5: End.

D. Attribute_Operation_Association Extraction

After obtaining the episodes, the terms are mapped to the result of the *Concept Clustering* to tag the concept name, as in the following example.

南韓(Nca|球隊),擊敗(VC),義大利(Nca|球隊)
Korea(Nca|team), beat(VC), Italy(Nca|team)

巴西(Nca|球隊),贏得(VJ3),冠軍(Nad|獎項)
Brazil(Nca|team), win(VJ3), champion (Nad|award)

英格蘭(Nca|球隊),隊長(Nab),貝克漢(Nba|球員)
England(Nca|team), captain(Nab), Beckham (Nba|team member)

李瓦度(Nba|球員),進球(VA)
Rivaldo (Nba|team member), goal(VA)

南韓隊(Nba|球隊),體能(Nad)
Korea (Nba|team), physical strength(Nad)

Herein, SOM approach is used to cluster concepts and instances, and the experts carry out the refining clustering so that the *Concept Clustering* result indicates that "南韓 (Korea)", "義大利 (Italy)", "巴西 (Brazil)", and "英格蘭 (England)" are determined to be an instance while "球隊 (team)" is determined to be a concept with both as nouns, "冠軍 (champion)" is an instance of the concept "獎項 (award)", and "貝克漢 (Beckham)" and "李瓦度 (Rivaldo)" are instances of concept "球員 (team member)". It is finished using a semi-automatic way and automatic concept clustering is the further study. The algorithm for mapping instances and concepts is shown below.

The notation for the mapping instances and the concept algorithm is as follows:

e_i: denotes an episode, where $1 \leqslant i \leqslant n$, and n denotes the number of all episodes.
t_j: denotes a term, where $1 \leqslant j \leqslant m$, and m denotes the number of all terms in an episode.

Mapping Instances and Concepts Algorithm
Input: All episodes and the result of *Concept Clustering*
Output: Episodes with the *concept_name* tag
Method:
Step 1: For all episodes e_i
  Step 1.1: For all terms t_i
    Step 1.1.1: If t_i denotes an instance
      Step 1.1.1.1: Obtain the *concept_name* of t_i from the concept clustering results
      Step 1.1.1.2: Transform $t_i(POS)$ into $t_i(POS|\ concept\_name)$
Step 2: End.

The morphological features of Chinese terms are now described. The Attributes, operations and associations are extracted from episodes according to the morphological information of the Chinese term and the Chinese syntax. For ontology construction, patterns such as "concept-attribute-value", "concept-association-concept" or "concept-operation" are extracted from the domain data. These patterns are treated as sentence patterns, such as "subject-verb-object" or "subject-modifier". However, the subject, object, and modifier are hard to extract from Chinese documents, because Chinese grammar is very complex. Therefore, the morphological features of Chinese terms were analyzed to assist the extraction of attributes, operations, and associations from

episodes. The *CKIP* of Academia Sinica classifies verbs into 12 categories. In the proposed morphological analysis, these 12 categories of verbs are classified into five groups according to their meanings and syntaxes. These five groups of verbs are treated as operations or associations by their morphological features, listed in Table 1. Operations describe actions of a concept, so verbs that only need a subject, are selected as operations. Associations describe relationship between two concepts, so verbs needing a subject and object, are selected as associations. Similarly, nouns are treated as concepts or properties, and are given attributes and associations according to their morphological features, as listed in Table 2. To reduce the complexity of the automated ontology construction, this study just discusses the similarities between concepts automatically created but validated by domain experts, and those between manual-created attributes and those between manual-created operations. That is, the attributes, operations, and associations are not totally extracted automatically using the proposed algorithm, but it needs the validation of the domain experts. Future study will discuss the similarities between automatic-constructed attributes or between automatic-constructed operations.

Finally, the *Attributes_Operations_Associations Extraction* algorithm is shown as follows:

The notation for the *Attributes_Operations_Associations Extraction* algorithm is given below.

e_i: denotes an episode, where $1 \leqslant i \leqslant n$ and n denotes the total number of episodes.

Attributes_Operations_Associations Extraction Algorithm
Input: Episodes with the *concept_name* tag
Output: Constructed domain ontology
Method:
Step 1: For all episodes e_i
  Step 1.1: If the number of terms in e_i is 2
    Step 1.1.1: If the first term t_1 is an instance, and the *POS* of the second term t_2 is Nab, Nac, Nad, Nae, Ncb or VH
      Step 1.1.1.1: The second term t_2 is an attribute of this instance t_1.

Table 1
Morphological analysis for Chinese verbs

Morphological feature	POS of CKIP	Description	Example	Role in ontology
Intransitive verb	VA	Only need subject	說話 "say", 進球 "goal"	Operation
Transitive verb	VB, VC, VD, VE, VF	Need subject and objective	擊敗 "beat", 預防 "prevent"	Association
Linking verb	VG	Link subject and objective and express a equivalent relation	等於 "equal", 尊稱 "name"	Association
Status intransitive verb	VH	Only need subject and describe status of subject	動聽 "to be pleasant to listen to", 炎熱 "hot"	Attribute
Status transitive verb	VI, VJ, VK, VL	Only need subject	造成 "cause", 遇到 "meet"	Association

Table 2
Morphological analysis for Chinese nouns

Morphological feature	POS of CKIP	Example	Role in ontology
Substance noun (uncountable concrete noun)	Naa	泥土 "soil", 雨 "rain"	Concept
Countable concrete noun	Nab	乘客 "passenger", 門將 "keeper"	Attribute, Association, Concept
Countable abstract noun	Nac	路徑 "path", 位置 "location"	Attribute, Association, Concept
Uncountable abstract noun	Nad	風度 "demeanor", 香氣 "aroma"	Attribute, Association, Concept
Collective noun	Nae	車輛 "vehicle", 獎金 "award"	Attribute, Association, Concept
Proper noun	Nb	雙魚座 "Pisces", 世足賽 "FIFA Word Cup"	Concept
Proper local noun	Nca	西班牙 "Spain", 台北 "Taipei"	Concept
Common local noun	Ncb	郵局 "pot office", 中心 "center"	Attribute, Association, Concept
A noun of locality	Ncc	海外 "abroad", 身上 "on the body"	Concept
Positional noun	Ncd	外頭 "outside", 左 "left"	Concept
Named local noun	Nce	四海 "over the word", 當地 "local"	Concept

Step 1.1.2: If the first term t_1 is an instance and the *POS* of the second term t_2 is VA
Step 1.1.2.1: The second term t_2 is an operation of this instance t_1.
Step 1.2: If the number of terms in e_i is 3
Step 1.2.1: If the first term t_1 and the third term t_3 are instances, and the *POS* of the second term t_2 is a transitive verb (VB, VC, VD, VE, VF), status transitive verb (VI, VJ, VK, VL), or Nab, Nac, Nad, Nae and Ncb.
Step 1.2.1.1: The second term t_2 is an association of the instances t_1 and t_3.
Step 2: Output the domain ontology.
Step 3: End.

3. A fuzzy inference mechanism for Chinese text ontology learning

This section introduces a parallel fuzzy inference mechanism to infer a new instance belonging to which one existing concept. Fuzzy logic is intended to alleviate difficulties in developing and analyzing complex systems encountered by conventional mathematical tools, based on the observation that human reasoning can adopt the concept and knowledge without well-defined, sharp boundaries (i.e., vague concepts) [33]. Therefore, fuzzy logic is suitable for natural language processing. The fuzzy inference mechanism is one of many methods to solve the natural language processing problem. However, previous studies [25,28,30] indicate that a new instance's concept can be easily inferred by parallel fuzzy inference. Therefore, this study adopts the fuzzy inference mechanism to infer a new instance. Future studies will try to infer a new instance using other methods. A new instance may contain many different attributes, operations and associations, which can be added to the concept to update the domain ontology. Fig. 5 shows the concept update process. Fig. 5(a) shows an existed concept in the domain ontology; Fig. 5(b) shows a new instance discovered from new documents, and Fig. 5(c) shows the updated concept of Fig. 5(a) and (b).

Section 3.1 describes the conceptual resonance between a concept and a new instance. Section 3.2 presents a parallel fuzzy inference mechanism for conceptual resonance computing.

3.1. Conceptual resonance between a concept and a new instance

The conceptual resonance is defined as a degree of belonging between a concept and a new instance. Hence, a new instance is likely to belong to a particular concept if the conceptual resonance between them is high. The conceptual resonance determines whether a new instance belongs to an existing or a new concept. The concept describes a group of instances with identical attributes, operations and associations to other instances. Therefore, if a new instance belongs to a new concept, then the instance and the concept have a strong conceptual resonance, and they probably have some identical attributes, operations and associations. If a new instance does not belong to any existing concepts in the domain ontology, then the instance and the concepts have a low mutual conceptual resonance, and they do not have many identical attributes, operations or associations. The conceptual resonance is inferred from existed concepts in the domain ontology.

This study adopts four fuzzy variables, *resonance strength in attribute* x_A, *resonance strength in operation* x_O, *resonance strength in association domain* x_D and *resonance strength in association range* x_R, to calculate the conceptual resonance strength between a concept and a new instance, which are described them as follows.

A. Resonance strength in attribute x_A

The term x_A denotes the ratio of identical attributes between an existing concept C of the domain ontology and a new instance I, and is calculated by Eq. (2):

$$x_A = \frac{\textit{the number of identical attributes in } C \textit{ and } I}{\textit{the number of attributes in } C} \tag{2}$$

This fuzzy variable defines two linguistic terms, *A_Low* and *A_High*. In this study, the trapezoidal function, shown in Eq. (3), is adopted as the membership function of linguistic terms and can be expressed as the parameter set $[a,b,c,d]$. For instance, the membership functions *A_Low* and *A_High*, can be denoted as $[0,0,0,0.6]$ and $[0,0.6,1,1]$, respectively.

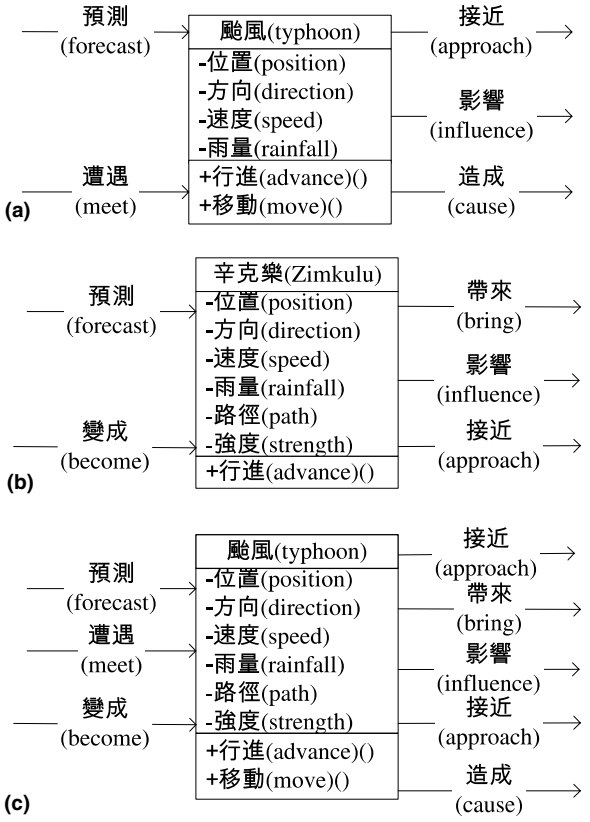


Fig. 5. The concept update process. (a) A concept in domain ontology. (b) A new instance discovered from new domain documents. (c) An updated concept after ontology learning.

$$f_{\text{trapezoidal}}(x:a,b,c,d)=\begin{cases}0 & x<a\\ (x-a)/(b-a) & a\leqslant x\leqslant b\\ 1 & b\leqslant x\leqslant c\\ (d-x)/(d-c) & c\leqslant x<d\\ 0 & x\geqslant d\end{cases} \tag{3}$$

For instance, in Fig. 5(a), the concept "颱風 (typhoon)" has four attributes in the domain ontology, namely "位置 (position)", "方向 (direction)", "速度 (speed)" and "降雨量 (rainfall)". Fig. 5(b) indicates that the number of attributes of the new instance "辛克樂 (Zimkulu)" is 6, and the attributes are "位置 (position)", "方向 (direction)", "速度 (speed)", "雨量 (rainfall)", "路徑 (path)", and "強度 (strength)". The concept "颱風 (typhoon)" and the new instance "辛克樂 (Zimkulu)" thus has four identical attributes between them, namely "位置 (position)", "方向 (direction)", "速度 (speed)" and "降雨量 (rainfall)." Hence, the value of this fuzzy variable $x_{\text{A}}=\frac{4}{4}=1$.

B. Resonance strength in operation x_O

The term x_{O} denotes the ratio of identical operations between an existing concept C of the domain ontology and a new instance I, and is calculated by Eq. (4):

$$x_{\text{O}}=\frac{\textit{the number of identical operations in } C \textit{ and } I}{\textit{the number of operations in } C} \tag{4}$$

The membership functions of x_{O} are the same as the fuzzy variable x_{A}, and the two linguistic terms *O_Low* and *O_High*, can be represented as $[0,0,0,0.6]$ and $[0,0.6,1,1]$, respectively. Fig. 5 indicates that the number of operations of the concept "颱風 (typhoon)" in the domain ontology is 2. The number of identical operations

between the concept "颱風 (typhoon)" and the new instance "辛克樂 (Zimkulu)" is 1. Therefore, the value of this fuzzy variable x_O is $\frac{1}{2} = 0.5$.

C. Resonance strength in association domain x_D

The term x_D denotes the ratio of identical association domain between an existing concept C of the domain ontology and a new instance I, and is calculated by Eq. (5):

$$x_D = \frac{\textit{the number of identical association domain in } C \textit{ and } I}{\textit{the number of association domain in } C} \tag{5}$$

An association domain is one in which the arrowhead of associations is toward other concepts or instances, such as the association among "帶來 (bring)," "影響 (influence)" and "接近 (approach)" in Fig. 5. This fuzzy variable defines three linguistic terms, namely *D_Low*, *D_Medium* and *D_High*, whose membership functions can be represented as $[0, 0, 0, 0.3]$, $[0, 0.3, 0.3, 0.5]$, and $[0.3, 0.5, 1, 1]$, respectively. As shown in Fig. 5, the number of the association domain of the concept "颱風 (typhoon)" in the domain ontology is 3. The number of the identical association domain between the concept "颱風 (typhoon)" and the instance "辛克樂 (Zimkulu)" is 2. Therefore, the value of this fuzzy variable x_D is $\frac{2}{3} = 0.67$.

D. Resonance strength in association range x_R

Term x_R denotes the ratio of identical association range between an existing concept C of domain ontology and a new instance I, and is calculated by Eq. (6):

$$x_R = \frac{\textit{the number of identical association range in } C \textit{ and } I}{\textit{the number of association range in } C} \tag{6}$$

The association range indicates that the arrowhead of associations is toward its own concept, such as associations "變成 (become)" and "遭遇 (meet)" in Fig. 5. The fuzzy variables x_R and x_D have the same membership functions, and also have three linguistic terms, *R_Low*, *R_Medium* and *R_High*, which are represented as $[0, 0, 0, 0.3]$, $[0, 0.3, 0.3, 0.5]$ and $[0.3, 0.5, 1, 1]$, respectively. As shown in Fig. 5, the number of association range of the concept "颱風 (typhoon)" in the domain ontology is 2. The number of the identical association range between the concept "颱風 (typhoon)" and the instance "辛克樂 (Zimkulu)" is 1. Therefore, the value of this fuzzy variable $x_R = \frac{1}{2} = 0.5$.

3.2. A parallel fuzzy inference mechanism for conceptual resonance computing

After describing the four fuzzy variables for calculating the conceptual resonance between an existing concept and a new instance, the parallel fuzzy inference architecture proposed by Lee et al. [25], Kuo et al. [7] and Lin and Lee [10] is used in this study. The structure comprises the premise layer, rule layer and conclusion layer. The model has two classes of node, fuzzy linguistic nodes and rule nodes. A fuzzy linguistic node denotes a fuzzy variable, and manipulates information related to a linguistic variable. A rule node denotes a rule, and determines the final firing strength of the rule during inference. The premise layer performs the first inference step to calculate the matching degrees. The conclusion layer is responsible for making conclusions and defuzzification. Each layer is described here in detail.

A. Premise layer

The first layer, called the premise layer, represents the premise part of the fuzzy inference system. Each fuzzy variable appearing in the premise part is represented by a condition node. Each output of the condition node is connected to some nodes in the second layer, forming a condition specified in some rules. The premise layer performs the first inference step to calculate degrees of matching. The input vector is given by $x = (x_1, x_2, \ldots, x_n)$, where x_i denotes the input value of linguistic node i. The output vector of the premise layer is thus

$$\mu^1 = ((u^1_{11}, u^1_{21}, \ldots, u^1_{N_1 1}), (u^1_{12}, u^1_{22}, \ldots, u^1_{N_2 2}), \ldots, (u^1_{1n}, u^1_{2n}, \ldots, u^1_{N_n n})) \tag{7}$$

where u^1_{ij} denotes the matching degree of linguistic term j in condition node i. The membership degree u^1_{ij} of the four fuzzy variables can be calculated by their membership functions described in Section 3.1.

B. Rule layer

Each node in the second layer, called the rule layer, is a rule node denoting a fuzzy rule. The links in this layer perform precondition matching of fuzzy logic rules, and the output of a rule node in the rule layer is linked with associated linguistic nodes in the third layer. In the proposed model, the rules are defined by the domain expert and are listed in Table 3. In the rule node, function f_r provides the net input for this node as in Eq. (8):

$$f_r = \sum_{i=1}^{N} \mu_i \tag{8}$$

Hence, this value f_r is not in a fixed range between 0 and 1. Therefore, a normalizing function S is adopted in this study. Function S is calculated by Eq. (9):

$$S(x:a,b) = \begin{cases} 0, & x < a \\ 2(\frac{x-a}{b-a})^2, & a \leqslant x \leqslant \frac{a+b}{2} \\ 1-2(\frac{x-b}{b-a})^2, & \frac{a+b}{2} \leqslant x < b \\ 1, & x \geqslant b \end{cases} \tag{9}$$

C. Conclusion layer

The third layer is called the conclusion layer, and also is composed of a set of fuzzy linguistic nodes. The output fuzzy variable in the proposed system is the conceptual resonance strength y_{CRS}, which denotes the conceptual resonance strength between an existing concept and a new instance. The four linguistic terms *CRS_Low*, *CRS_Medium_Low*, *CRS_Medium_High*, and *CRS_High* can be defined in $[0,0,0,0.3]$, $[0,0.3,0.3,0.6]$, $[0.3,0.6,0.6,0.9]$, and $[0.6,0.9,1,1]$, respectively. Defuzzification may be needed at the end of the inference process. The final output y of the proposed approach is the crisp value produced by combining all inference results with their firing strength. Eq. (10) presents the defuzzification formula:

$$y = \frac{\sum_{i=1}^{r}\sum_{j=1}^{c} y_{ij}^{k} w_{ij}^{k} V_{ij}}{\sum_{i=1}^{r}\sum_{j=1}^{c} y_{ij}^{k} w_{ij}^{k}} \tag{10}$$

where $w^k = \frac{\sum_{i=1}^{n} \mu_i^1}{n}$; V_{ij} denotes the center of gravity; r denotes the numbers of corresponding rule nodes; c denotes the number of linguistic terms in the output node; n denotes the numbers of the fuzzy variable in the

Table 3
The Fuzzy rules table

Rule	x_A	x_O	x_D	x_R	y_{CRS}	Rule	x_A	x_O	x_D	x_R	y_{CRS}
1	L	L	L	L	L	**19**	H	L	L	L	ML
2	L	L	L	M	L	**20**	H	L	L	M	ML
3	L	L	L	H	L	**21**	H	L	L	H	ML
4	L	L	M	L	L	**22**	H	L	M	L	ML
5	L	L	M	M	L	**23**	H	L	M	M	ML
6	L	L	M	H	ML	**24**	H	L	M	H	MH
7	L	L	H	L	L	**25**	H	L	H	L	ML
8	L	L	H	M	ML	**26**	H	L	H	M	MH
9	L	L	H	H	ML	**27**	H	L	H	H	MH
10	L	H	L	L	L	**28**	H	H	L	L	ML
11	L	H	L	M	ML	**29**	H	H	L	M	MH
12	L	H	L	H	ML	**30**	H	H	L	H	MH
13	L	H	M	L	ML	**31**	H	H	M	L	MH
14	L	H	M	M	ML	**32**	H	H	M	M	MH
15	L	H	M	H	MH	**33**	H	H	M	H	H
16	L	H	H	L	ML	**34**	H	H	H	L	MH
17	L	H	H	M	MH	**35**	H	H	H	M	H
18	L	H	H	H	MH	**36**	H	H	H	H	H

Table 4
Conceptual resonance between two terms

A new instance and an existed concept	Conceptual resonance	
辛樂克 (Zimkulu), 氣流 (airstream)	0.525779958	
辛樂克 (Zimkulu), 雨 (rain)	0.487075105	
辛樂克 (Zimkulu), 颱風 (typhoon)	0.650331524	*
辛樂克 (Zimkulu), 災害 (calamity)	0.497960395	

premise layer, and k denotes the current layer number. The values of r, c, n, and k adopted in this study are 36, 4, 4, and 2, respectively.

Table 4 shows an example of the conceptual resonance between a new instance "辛克樂 (Zimkulu)" and existing concepts "氣流 (airstream)", "雨 (rain)", "颱風 (typhoon)" and "災害 (calamity)". The value of conceptual resonance between "辛樂克 (Zimkulu)" and "颱風 (typhoon)" is the highest, so the inference mechanism infers that the new instance "辛樂克 (Zimkulu)" belongs to the concept "颱風 (typhoon)".

Therefore, a new instance belongs to an existing concept with the highest y_{CRS}. If all y_{CRS} values are smaller than the threshold, then the new instance must either generate a new concept or be discarded by the domain experts.

4. Experimental result

This study adopted the Chinese 2002 FIFA (Federation Internationale de Football Association) World Cup news and the typhoon news as the domain data to construct the *2002 FIFA World Cup ontology* and the *typhoon ontology*. In the experiments, the input data were divided into two parts, namely training and testing data. The experimental results for these two domains are listed herein. The *2002 FIFA World Cup domain*

Table 5
Result of *SOM* clustering in the 2002 FIFA World Cup domain

Concept name	Instances
比賽 (game)	四強 (semifinals), 十六強 (round of 16), 八強 (quarterfinal), 世足賽 (FIFA word cup), 冠軍賽 (final), 預賽 (preliminary contest), 小組賽 (group match), 準決賽 (semifinal), ...
組織 (organization)	足聯 (Federation Internationale de Football Association), 球會 (football society), 足協 (football association), ...
球隊 (team)	南韓隊 (Korea), 英格蘭隊 (England), 德國隊 (Germany), 日本隊 (Japan), 巴西隊 (Brazil), 義大利隊 (Italy), 喀麥隆 (Cameroon), 塞內加爾 (Senegal), ...
獎項 (award)	金杯 (World Cup), 金靴獎 (Golden Shoe), 金球獎 (Golden Ball), 冠軍 (champion), ...
球員 (team member)	蘇克 (SUKER), 羅納迪諾 (RONALDINHO), 羅納度 (RONALDO), 李瓦度 (RIVALDO), 李雲在 (LEE Woon Jae), 卡洛斯 (ROBERTO), 卡恩 (KAHN), 歐文 (OWEN), 席丹 (ZIDANE), 貝克漢 (BECKHAM), ...
裁判 (referee)	莫雷諾 (MORENO), 科利納 (KOLLINA), 莫瑞 (MOURAD)
教練 (coach)	沃勒 (VOLLER), 艾利克森 (ERIKSSON), 卡馬喬 (CAMACHO), 希丁克 (HIIDDINK), 斯科拉里 (SCOLARI), ...
規則 (rule)	紅牌 (red card), 黃牌 (yellow card), ...
設備 (equipment)	球門 (goal), 球柱 (goalpost), 球門區 (goal-area), ...

Table 6
Results of *Attributes_Operations_Associations Extraction* in 2002 FIFA World Cup domain

	Attributes	Operations	Associations	Associations of instances
$Min = 3$, $Win = 10$	242	84	112	190
$Min = 5$, $Win = 10$	74	32	16	28
$Min = 7$, $Win = 10$	34	15	5	9

Table 7
Precision of *Attributes_Operations_Associations Extraction* judged by domain experts in 2002 FIFA World Cup domain

	Attributes (%)	Operations (%)	Associations (%)	Associations of instances (%)	Average (%)
$Min = 3,\ Win = 10$	31.82	47.62	31.25	36.84	36.88
$Min = 5,\ Win = 10$	37.84	59.38	68.75	82.14	62.03
$Min = 7,\ Win = 10$	32.35	60	100	100	73.09

had 879 documents, of which 440 were placed in the training data, and 439 in the testing data. Table 5 lists the *Concept Clustering* results by the *SOM*. Table 6 lists the results of the *Attributes_Operations_Associations*

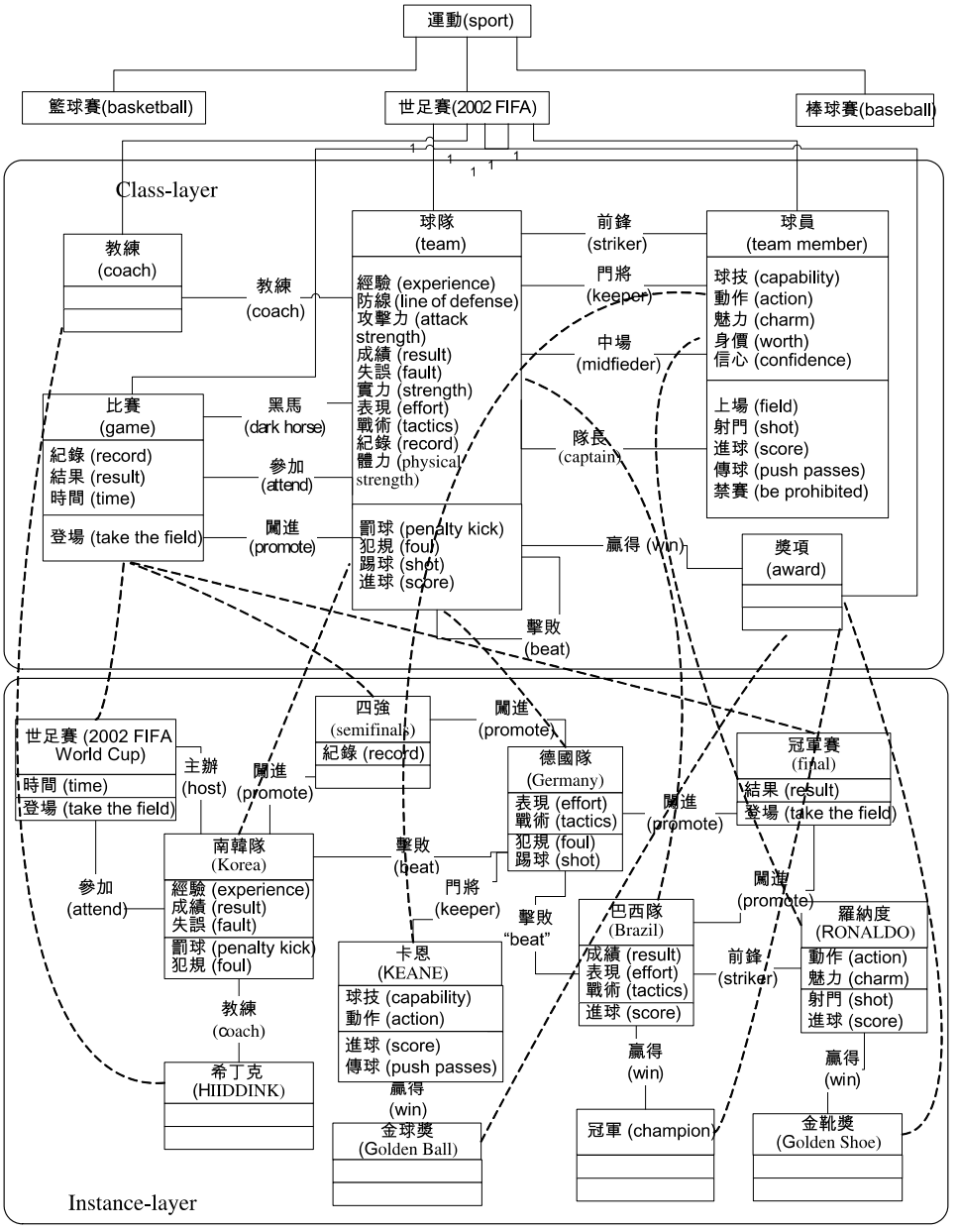


Fig. 6. Part of the constructed *2002 FIFA World Cup ontology*.

Table 8
Results of Attributes_Operations_Associations Extraction in typhoon domain

	Attributes	Operations	Associations	Associations of instances
$Min = 3,\ Win = 10$	59	11	259	366
$Min = 5,\ Win = 10$	28	4	101	137
$Min = 7,\ Win = 10$	17	4	49	71

Table 9
Precision of *Attributes_Operations_Associations Extraction* judged by domain experts in *typhoon domain*

	Attributes (%)	Operations (%)	Associations (%)	Associations of instances (%)	Average (%)
$Min = 3,\ Win = 10$	38.98	54.55	21.24	25.96	35.18
$Min = 5,\ Win = 10$	50	75	29.70	32.28	46.75
$Min = 7,\ Win = 10$	64.71	75	53.06	50.70	60.87

Extraction. The *Min* and *Win* are the minimum occurrence and window size, respectively. Table 7 lists the precisions of the *Attributes_Operations_Associations Extraction* judged by domain experts. Fig. 6 shows a part of the constructed *2002 FIFA World Cup ontology*.

The typhoon domain had a total of 185 documents, of which 93 documents were placed in the training data, and 92 in the test data. Table 8 lists the results of the *Attributes_Operations_Associations Extraction*. Table 9 shows the precision of the *Attributes_Operations_Associations Extraction* judged by domain experts. Experimental results indicate that the larger the training data set, the better the precision results for attributes, operations and associations. Although the result is not perfect, some intermediate results of this approach can help domain experts validate the domain ontology and discover further domain knowledge.

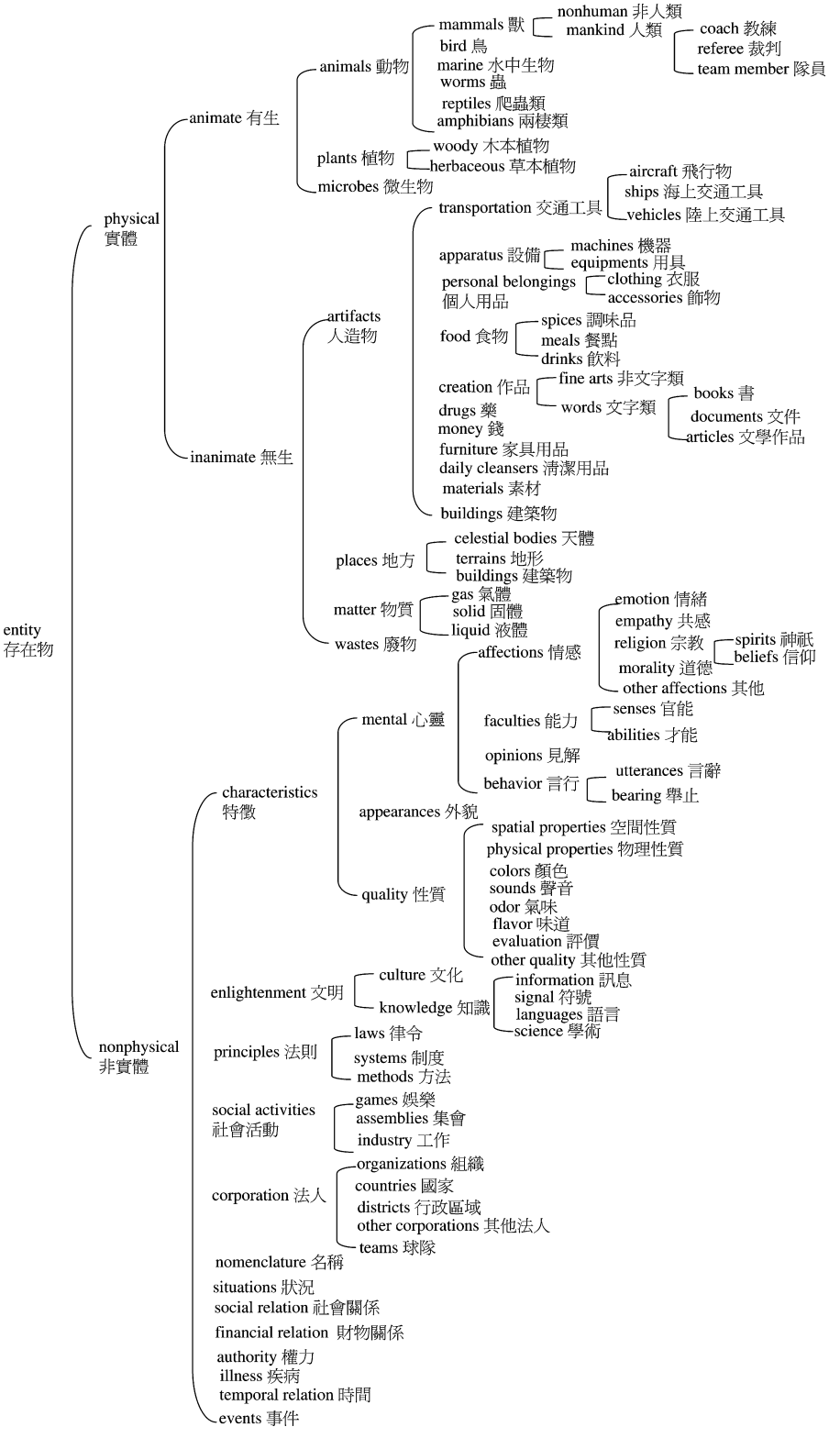
5. Conclusions and future work

This study proposes an episode-based fuzzy inference mechanism to extract domain ontology from unstructured Chinese text documents. After finishing the domain ontology construction, the domain expert is required to validate and correct the generated ontology. Ontology editors and other toolkits are also necessary for the whole process of automated ontological construction and maintenance. Additionally, the *SOM* algorithm was adopted for concept clustering and defining taxonomic relationships. The attributes and operations of concepts can be extracted based on ontology construction episodes. The non-taxonomic relationships are also generated from episodes. Moreover, the fuzzy inference mechanism is adopted to obtain new instances for the ontology learning. Experimental results indicate that the proposed approach can successfully construct the Chinese text domain ontology. However, for some special cases, such as a domain with rapid changing terms and concepts or with complex semantics, it is seen as an area of application for automated ontology construction – consider for example the potential of automated ontology construction for query expansion in information retrieval in the news domain. In the opposite, if a very carefully and accurate designed ontology is needed, an ontology engineer might be quicker to do that by hand – for example, an ontology describing the functions and behaviors of an airplane might possibly be constructed manually rather than generated from the airplanes documentation. Future work will include efforts to improve the precision of the proposed method, and studying the learning mechanism for fuzzy inference rules. The proposed approach will also be applied to other languages with semantic corpus or semantic dictionaries such as *CKIP*.

Acknowledgements

The authors would like to thank the anonymous referees for their constructive and useful comments. This study is partially sponsored by Department of Industrial Technology, Ministry of Economic Affairs, R.O.C. under the grant 95-EC-17-A-02-S1-029 and partially supported by the National Science Council of Taiwan under the Grant NSC94-2213-E-024-006.

Appendix. Conceptual structure in *CKIP* (modified for 2002 FIFA domain)



References

[1] H. Ahonen, O. Heinonen, M. Klemettinen, A.I. Verkamo, Applying data mining techniques for descriptive phrase extraction in digital document collections, in: Proceedings of the Advances in Digital Libraries Conference, Santa Barbara, CA, 1998, pp. 2–11.
[2] A. Farquhar, R. Fikes, J. Rice, The Ontolingua Server: a tool for collaborative ontology construction, International Journal of Human–Computer Studies 46 (6) (1997) 707–727.
[3] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, UK, Harlow, 1999.
[4] T. Gruber, What is an Ontology?, URL Accessed on November 9, 2001. Available from: <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>.
[5] L. Khan, F. Luo, Ontology construction for information selection, in: Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence, Crystal City, Virginia, 2002, pp. 122–127.
[6] T. Kohonen, Self-Organizing Maps, second ed., Springer-Verlag, Heidelberg, 1997.
[7] Y.H. Kuo, J.P. Hsu, C.W. Wang, A parallel fuzzy inference model with distributed prediction scheme for reinforcement learning, IEEE Systems, Man, and Cybernetics 28 (2) (1998) 160–172.
[8] C.S. Lee, C.H. Liao, Y.H. Kuo, A semantic-based concept clustering mechanism for Chinese news ontology construction, in: Workshop on Artificial Intelligence of International Computer Symposium, Taiwan, 2002.
[9] D.B. Lenat, CYC: a large-scale investment in knowledge infrastructure, Communications of the ACM 38 (11) (1995) 33–41.
[10] C.T. Lin, C.S.G. Lee, Neural-network-based fuzzy logic control and decision system, IEEE Computers 40 (12) (1991) 1320–1336.
[11] A. Maedche, S. Staab, Ontology learning for the semantic web, IEEE Intelligent Systems 16 (2) (2001) 72–79.
[12] H. Mannila, H. Toivonen, A.I. Verkamo, Discovery of frequent episodes in event sequences, International Journal of Data Mining and Knowledge Discovery 1 (3) (1997) 259–289.
[13] G.A. Miller, WORDNET: an on-line lexical database, International Journal of Lexiography 3 (4) (1990) 235–312.
[14] M. Missikoff, R. Navigli, P. Velardi, Integrated approach to web ontology learning and engineering, IEEE Computer 35 (11) (2002) 60–63.
[15] R. Navigli, P. Velardi, A. Gangemi, Ontology learning and its application to automated terminology translation, IEEE Intelligent Systems 18 (1) (2003) 22–31.
[16] N.F. Noy, M. Sintek, S. Decker, M. Crubezy, R.W. Fergerson, M.A. Musen, Creating semantic web contents with Protege-2000, IEEE Intelligent Systems 16 (2) (2001) 60–71.
[17] V. Sugumaran, V.C. Storey, Ontologies for conceptual modeling: their creation, use, and management, International Journal of Data and Knowledge Engineering 42 (3) (1997) 251–271.
[18] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer, D. Wenke, OntoEdit: collaborative ontology development for the semantic web, in: Proceedings of the First International Semantic Web Conference, Sardinia, Italia, 2002.
[19] V.W. Soo, C.Y. Lin, Ontology-based information retrieval in a multi-agent system for digital library, in: Proceedings of the Sixth Conference on Artificial Intelligence and Applications, Taiwan, 2001, pp. 241–246.
[20] D.H. Widyantoro, J. Yen, A fuzzy ontology-based abstract search engine and its user studies, in: Proceedings of the 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, 2001.
[21] S.J. Yen, A.L.P. Chen, An efficient approach to discovering knowledge from large databases, in: Proceedings of the 4th International Conference on Parallel and Distributed Information Systems, Florida, United States, 1996, pp. 8–18.
[22] K. Yoshinaga, T. Terano, N. Zhong, Multi-lingual intelligent information retriever with automated ontology generator, in: Proceedings of the Third International Conference on Knowledge-Based Intelligent Information Engineering Systems, Adelaide, Australia, 1999, pp. 62–65.
[23] L. Zhou, Q.E. Booker, D. Zhang, ROD – toward rapid ontology development for underdeveloped domains, in: Proceedings of the 35th Annual Hawaii International Conference on System Sciences, 2002.
[24] Academia Sinica, Chinese Electronic Dictionary, in: Technical Report 93-05, Taiwan, 1993.
[25] C.S. Lee, Y.H. Kuo, C.H. Liao, Z.W. Jian, A Chinese term clustering mechanism for generating semantic concepts of a news ontology, International Journal of Computational Linguistics and Chinese Language Processing 10 (2) (2005) 277–302.
[26] C.S. Lee, C.C. Jiang, T.C. Hsieh, A genetic fuzzy agent using ontology model for meeting scheduling system, Information Sciences 176 (9) (2006) 1131–1155.
[27] T. Andreasen, P.A. Jensen, J.F. Nilsson, P. Paggio, B.S. Pedersen, H.E. Thomsen, Content-based text querying with ontological descriptors 48 (2004) 199–219.
[28] C.S. Lee, Z.W. Jian, L.K. Huang, A fuzzy ontology and its application to news summarization, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 35 (5) (2005) 859–880.
[29] N. Lammari, E. Metais, Building and maintaining ontologies: a set of algorithm, Data and Knowledge Engineering 48 (2004) 155–176.
[30] C.S. Lee, Y.J. Chen, Z.W. Jian, Ontology-based fuzzy event extraction agent for Chinese e-news summarization, Expert Systems with Applications 25 (3) (2003) 431–447.
[31] N. Guarino, C. Masolo, G. Vetere, OntoSeek: content-based access to the web, IEEE Intelligent Systems 14 (3) (1999) 70–80.
[32] L. Khan, F. Luo, Hierarchical clustering for complex data, International Journal on Artificial Intelligence Tools 14 (5) (2005) 1–19.
[33] J. Yen, R. Langari, Fuzzy Logic, Prentice-Hall, Inc., New Jersey, 1999.
[34] D. Elliman, J. Rafael, G. Pulido. Automatic derivation of on-line document ontology, MERIT 2001, 15th European Conference on Object Oriented Programming, Budapest, Hungary, 2001.
[35] A. Hotho, A. Madche, S. Staab, Ontology-based text clustering, in: Proceedings of the IJCAI-2001 Workshop Text Learning: Beyond Supervision, Seattle, USA, 2001, pp. 48–54.

[36] M. Horridge, H. Knublauch, A. Rector, R. Stevens, C. Wroe, A Practical Guide to Building OWL Ontologies Using the Protege-OWL Plugin and CO-ODE Tools Edition 1.0, URL Accessed on August 27, 2004. Available from: <http://www.co-ode.org/resources/tutorials/ProtegeOWLTutorial.pdf>.
[37] A.P. Azcarraga, M.H. Hsieh, S.L. Pan, R. Setiono, Extracting salient dimensions for automatic SOM labeling, IEEE Transactions on SMC – Part C 35 (4) (2005) 595–600

Chang-Shing Lee received the B.S. degree in Information and Computer Engineering from the Chung Yuan Christian University, Chung-Li, Taiwan, in 1992, and the M.S. degree in Computer Science and Information Engineering from the National Chung Cheng University, Chia-Yi, Taiwan, in 1994, and the Ph.D. degree in Computer Science and Information Engineering from the National Cheng Kung University, Tainan, Taiwan, in 1998. From August 2001 to July 2003, he jointed the faculty of the Department of Information Management, Chang Jung Christian University as an Assistant Professor. He became an Associate Professor in the Department of Information Management, Chang Jung Christian University since August 2003. Now he is currently an Associate Professor in the Department of Computer Science and Information Engineering, National University of Tainan, Taiwan. His research interests include intelligent agent, ontology engineering, knowledge management, Web services, semantic Web, and soft computing systems. He holds several patents on ontology engineering, document classification, and image filtering. Dr. Lee received the MOE's Campus Software Award in 2002, the CJCU's Outstanding Research Achievement Award in 2003, the Outstanding Teacher Award from Chang Jung Christian University in 2004, and the TAAI Advisor's Award in 2005. He has guest edited a special issue for Journal of Internet Technology. He is a Member of TAAI.

Yuan-Fang Kao received the B.S. degree in Information and Computer Engineering from the Chung Yuan Christian University, Chung-Li, Taiwan, in 2001, and the M.S. degree in Computer Science and Information Engineering from the National Cheng Kung University, Tainan, Taiwan, in 2003. Her research interests are ontology development and applications, object-oriented modeling, knowledge acquisition and Semantic Web.

Yau-Hwang Kuo received his Ph.D. in Electrical Engineering from the National Cheng Kung University (NCKU), Tainan, Taiwan, in 1988. He is on the faculty of the Department of Computer Science and Information Engineering at NCKU. He is now the Director of the CREDIT Research Center at the NCKU, and the Standing Board of Taiwan AI Society, Chinese Fuzzy Society and Object-Oriented Technology SIG. His research interests include neural network, fuzzy logic, knowledge engineering, Internet multimedia communication, Internet information retrieval and Digital VLSI design.

Mei-Hui Wang received the B.S. degree in BioMedical Engineering from the Chung Yuan Christian University, Chung-Li, Taiwan, in 1993, and the M.S. degree in Electrical Engineering from the Yuan Ze University, Chung-Li, Taiwan, in 1995. From July 1995 to June 2005, she worked for the Delta Electronics, Inc., Chung-Li, Taiwan, as a senior firmware engineer. Now she is currently a researcher in the Department of Computer Science and Information Engineering, National University of Tainan, Taiwan. Her research interests include intelligent agent, ontology engineering, and image processing.