# Domain Event Extraction and Representation with Domain Ontology

Shih-Hung Wu, Tzong-Han Tsai and Wen-Lian Hsu
Institute of Information Science
Academia Sinica
Nankang, Taipei, Taiwan, R.O.C.
shwu@iis.sinica.edu.tw, thtsai@iis.sinica.edu.tw, hsu@iis.sinica.edu.tw

## Abstract

With domain ontology, a meaningful index of document indexing, such as the domain events structure in this paper, can be defined. Since the construction of domain ontology is costly, an automatic domain ontology acquisition is preferred. We find that named entity can be a clue of domain event acquisition from the training corpus. Therefore, we add the named entity recognition module into the automatic domain ontology acquisition process. Thus, a subject-verb-object-modifier (SVOM) indexing can be constructed. Our experimental results demonstrate that the automatic event extraction and ontology acquisition can be good resources for text categorization and further information processing.

## 1 Introduction

Resources are very important for natural language processing (NLP) and information retrieval (IR). With the help of domain ontology, we can understand the relationships among the concepts in a text, and use this knowledge in various applications. [Gruber, 1993] However, updating the domain ontology with new terms is crucial when dealing with contemporary dynamic data.

In previous work, we built an event structure framework Information Map (InfoMap) and a semi-automatic tool for domain ontology acquisition (SOAT) [Hsu et al., 2001][Wu et al., 2002]. SOAT can automatically collect new concepts and recognize new relationships between concepts in a domain using linguistic templates. SOAT can only handle static data, but dynamic data is necessary some applications. The dynamic data are lists of domain concepts, named entity such as names of people, place, organization, expressions of time/date, numbers, expressions of quantity, and addresses. Due to the issues of size and manageability, items of dynamic data normally are not stored in domain ontology. Thus, we have added a new module to InfoMap to process dynamic data, including personal data, names, places, dates, etc.

With the help of information extraction, we can collect dynamic terms and link them with concepts in the ontology.

Thus, we combine existing ontologies with dynamic databases and create a new resource for NLP. [Jacquemin & Tzoukermann 1999] [Smeaton, 1999]

Text categorization is used to test the power of this combined resource. The traditional approach to text categorization uses only the training corpus to build a classifier. However, since there are always new concepts and terms in the data, a static resource does not always provide good results. A resource that can be maintained independently and updated dynamically is a significant improvement.

Traditional IR uses keywords or implicit rules, such as latent semantic indexing, to index a text. However, humans recognize a text through key concepts. Therefore, we believe that an event structure would be a better index unit. An event structure consists of a topic and its attribute or action. For example, for the keyword "car", "car sales" is a business event and "car racing" is a sports event.

One drawback to event structure indexing is the lexicon problem; there are many lexicons (for example human name, organization name, address and time expression) that should not be part of the domain ontology. We collect the lexicons in a separate database and link the mapping of lexicons to the concept in domain ontology.

For news story categorization, we use the collocation of human names and activities as important clues. For example, if a news article contains the sentence "President Chen Shui-bian attends the joint graduation ceremony of five military schools," the article should be classified in the "Politics" category. We know that President Chen Shui-bian is a politician and that most newsworthy people regularly appear in a specific news category. "Attend" is a verb and "the joint graduation ceremony" is its object, while "five military schools" is the modifier. If we only look at the subject and verb, "President Chen Shui-bian attends", we cannot determine the appropriate category. Nor can we categorize it correctly if we take only "President Chen Shui-bian attends ceremony." However, if we consider "the joint graduation ceremony of five military schools" and use the fact that "military schools" is in the military category, we can correctly determine that this story should be categorized as military news as well.
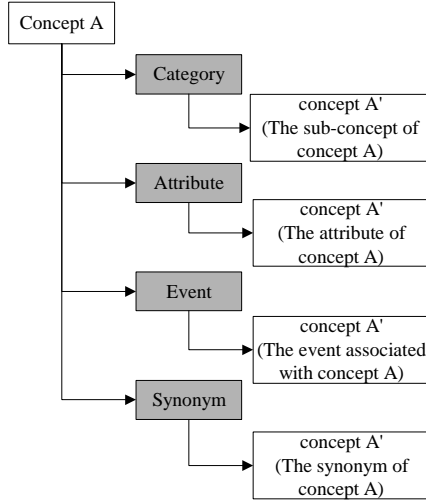
**Figure 1. Ontology format of InfoMap**

Consider another sentence, "President Chen Shui-bian attends the victory ceremony of the 21st Asian Cycling Championships." Clearly, this article should be categorized as Sports news since the modifier is "the 21st Asian Cycling Championships."

From these examples, we know that taking the collocation of subject, verb, object, and modifier, i.e., the structure of a sentence; we can sort a news story into the proper categories.

In this paper, we introduce our method for identifying event structures from text with domain ontology as our NLP resource. We also describe how we update the domain ontology to work with our new dynamic database.

## 2 Event Structure

There is usually a concrete semantic unit in a sentence when the sentence carries important domain information. Therefore, we can often represent a sentence by identifying the event structure in the sentence.

The event structure is defined as a domain concept pair that has strong semantic relationship defined in the ontology. For example, a concept and its attribute or action such as "car racing" or "car sales", can be event structures. Since we can perform a hypernym-hyponym substitution, the event structure can be generalized to "vehicle-sport" or "vehicle-merchandize" by an inference engine.

Domain concepts and the associated attributes or actions can be acquired from the domain corpus automatically. Thus, identifying the domain event structure is a useful knowledge management tool. However, certain domains, such as a news report about technology might have many new concepts and words and therefore, should be treated differently.

### 2.1 InfoMap Format

As a general representation of domain ontology, InfoMap consists of domain concepts, their related sub-concepts such
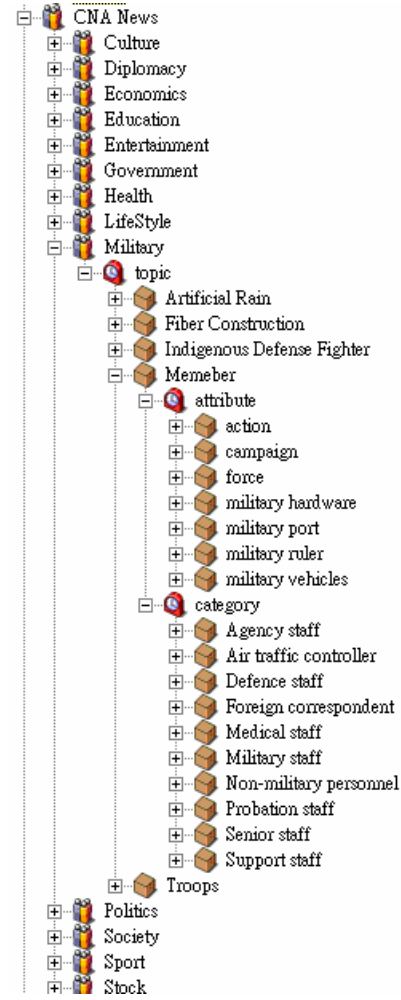


**Figure 2. Ontology Structure for CNA News**

as categories, attributes and actions. The relationship of a concept to its associated sub-concepts forms a tree-like taxonomy. InfoMap not only classifies concepts, but also connects the concepts by defining their relationships. Figure 1 shows a concept's skeleton created by InfoMap.

In InfoMap, concept nodes represent concepts and function nodes represent the relations between concepts. The root node of a domain is the name of the domain. Following the root node, we store the important topics in domain. These topics have sub-categories that recursively list related sub-topics. Figure 2 is only a partial view of the domain ontology of the CNA. Under each domain are several topics; each topic might have sub-concepts and associated attributes. Note that in this example, the domain ontology is automatically acquired from a domain corpus. This ontology is used in our experiment.

### 2.2 Event-Structure Based Document Indexing

The traditional information retrieval method represents a document as a vector of terms or n-grams that facilitate

full-text search. However, indexing a document with an event-structure provides a higher level of understanding of the document. Domain ontology can identify key sentences of the document and report the events. Thus, it serves as a basis for text summarization, question answering, knowledge management and text categorization.

# 3 Combine Ontology and Contemporary Database

Human names are very important in the news report domain, because news stories are generally categorized according to the names in the stories. The associated domain action often follows the domain key person, such as "Bush gives a speech" or "Jordan played a game". Since there might be numerous names in a domain and the fact that they change rapidly, the names themselves should not be part of domain ontology and should be stored in database. The combination of such a database with domain ontology can be viewed as a new language resource. We store the name of politicians in a database for political news categorization.

## 3.1 Domain Name Recognition

There are several studies that focus on named entity recognition, and provide useful heuristics. [Thompson and Dozier, 1999] The following represents a simple heuristic: find the common family names and treat the following word as the given name (Chinese name convention). Heuristics provide good recall for human name candidates finding. With the help of search engine, we can further confirm the names and gives good precision.

Our name entity recognition system for Chinese consists of two parts: a name candidate collector and a name checker. We start with a database of about 400 Chinese family names. When we come across a character or a string of characters in this database, we extract the following first and second characters. After collecting candidates, we check the names in the testing set as follows. First, we query the human name database. If the name exists in the database, it is selected. If not, we use the following heuristic to check if the name candidate is a valid personal name. We create the following strings and send them to a search engine.
1. Mr. + Name Candidate
    Ex: "Mr. Chen Shui-bian"
2. Miss + Name Candidate
    Ex: "Miss Jane Chen"
3. Mrs. + Name Candidate
    Ex: "Mrs. Eugenia Chen"
If we receive positive results from a search engine, we can be certain that the name exists.

If these search strings yield no results, we send the name again without the title. If the search engine returns results, we check to see if the string appears alongside other names and between a pair of quotation marks or commas. If it does, we regard the candidate as a name.

## 3.2 Associated Action Extraction

We find the associated action of domain names (those in database and those newly extracted) by examining the training corpus. If a name in a sentence is followed by a verb, then the verb is the candidate of domain action. We then check the verb in the NVEF database to find the suitable object for the verb. Thus, a Subject-Verb-Object (SVO) type of domain event structure candidate is extracted. Where NVEF is a database containing 400,000 Chinese verb-noun pairs that appear in a large corpus. [Tsai et al., 2002]

## 3.3 Modifier Extraction

As we mentioned in section 1, the object modifiers are also important. Therefore, we locate object modifiers by examining the training corpus and collect all the words that appear between the verbs and objects pairs. We then use a pruning step to filter out the candidates.

After modifier extraction, we calculate the likelihood that modifiers will collocate with object nouns. Suitable modifiers become children nodes of the object noun nodes. We crosscheck to ensure that all complete leaf node paths are found in the training corpus. If found, we leave the path on the InfoMap. Next, we calculate the Chi-squire value of all leaf node paths. If the Chi-squire values of any paths are under the threshold, we discard the extra leaf nodes. Thus, Thus, a Subject-Verb-Object-Modifier (SVOM) type of domain event structure candidate is extracted. Since we can generalize the S or O, the indexing power is quite rich.

## 3.4 Potential Applications

One advantage of the domain ontology as the domain knowledge representation is that it can serve many applications. We can construct domain ontology for one application and reuse it for other applications that use the same domain knowledge. For example, a domain ontology can be used for text categorization, question answering and document summarization.

Consider the military domain ontology in Figure 2. For a text categorization application, if an article mentions about the topics: "artificial rain", "fiber construction" or "indigenous defense fighter", then it can be classified in the military domain. For a question answering application, if a query is about " the agency staff action" or "the air traffic controller campaign", then they all can be generalized to "military/member-attribute" event structure. Since "the agency staff" and "the air traffic controller" are children nodes of "military/member", while "action" and "campaign" are children nodes of "military/member/attribute". For a document summarization application, we can only keep the sentences that contain domain event structures. Since domain event contains domain keywords and associated domain events, it must be a sentence that carries important domain information.

## 4 Experiment

We conduct a text categorization experiment to evaluate the capability of document indexing of event structure to categorize text.

We collect daily news stories from China News Agency (CNA). The news stories ranged from 1991 to 1999. Each of the news stories is short with, on average, 352 Chinese characters only (about 150 words). Originally there is more than thirty domains. Since the boundary between some domains is not well defined, we choose 12 major categories for our test. The 12 categories are Domestic Business (BD), Domestic Arts and Education (DD), Foreign Affairs (FA), Domestic Finance (FD), Domestic Health (HD), Taiwan local news (JD), Taiwan sports (LD), Domestic Military (MD), Domestic Politics (PD), Taiwan Stock Markets (SD), Domestic Travel (TD) and Weather Report (WE). From each category, we choose the first 200 news stories as our training set and the following 200 news stories as our testing set. Domain ontologies are acquired automatically from the training set. Our results for news categorization are shown in table 1.

### 4.1 News Categorization

Upon receiving a news story C, we separate it into sentences $S_i$. The sentences are scored and categorized according to domains. Thus, each sentence has an individual score for each domain $Score(D, S_i)$. We add up the score for each sentence in each domain, and compare the scores. The domain that has the highest score is the domain into which the text is categorized.

$$Domain(C) = \arg\max_D (\sum_{S \in C} SimScore(D, S_i))$$

The similarity score *SimScore* is defined according to the domain speculation process in the following subsection.

### 4.2 Domain Speculation

The goal of domain speculation is to categorize a sentence $S$ into a domain $D_j$, according the combined score of the keywords and the event structure in sentence $S$. We first calculate the similarity score of $S$ and $D_j$

$$SimScore(D_j, S) = Keyword\_Score(D_j, S) +$$
$$a * EventStructure\_Score(D_j, S)$$

where the keyword score and the event structure score are calculated independently.

We use the TFIDF classifier to calculate the *Keyword_Score* of a sentence as follows. First, we use a segmentation module to separate a Chinese sentence into words. The TFIDF classifier represents a domain as a weighted vector $D_j = ( w_{j1}, w_{j2}, \ldots, w_{jn})$ where $n$ is the number of words in this domain and $w_k$ is the weight of word $k$, $w_k$ is defined as $nf_{jk} * idf_{jk}$, where $nf_{jk}$ is the term frequency, (i.e., the number of times the word $w_k$ occurs in the domain $j$). Let $DF_k$ be the number of domains in which word $k$ appears and $|D|$ the total

number of domains. The inverse document frequency $idf_k$ is given by:

$$idf_k = \log(\frac{|D|}{DF_k}) \cdot$$

This weighting function assigns high values to domain-specific words, i.e., words that appear frequently in one domain and infrequently in others. Conversely, it assigns low weight to words that appear in many domains. The similarity between a domain $j$ and a sentence represented by a vector $D_i$ is measured by the cosine

$$Keyword\_Score(D_j, S)$$
$$= Sim(D_j, D_i)$$
$$= \frac{\sum_{k=1}^{n} w_{jk} w_{ik}}{\sqrt{\sum_{k=1}^{n}(w_{jk})^2 \sum_{k=1}^{n}(w_{ik})^2}}$$

The event structure score is calculated by the InfoMap engine. First, find all the nodes in an ontology that match the words in the sentence. Then, determine if there is any concept-attribute pair, or hypernym-hyponym pair. Finally, assign a score to each fired event structure according to the string length of words that match the nodes in the ontology. The selected event structure is the one with the highest score.

$$EventStructure\_Score(D_j, S)$$
$$= \max_{Event} \sum StringLength(keywords(D_j \cap S))$$

**Table 1. Experiment result of CNA news categorization.**

| Domain | Precision | Recall | F-Score |
|---|---|---|---|
| BD | 78.72% | 92.50% | 85.06% |
| DD | 83.02% | 88% | 85.44% |
| FA | 85.71% | 84% | 84.85% |
| FD | 97.58% | 80.50% | 88.22% |
| HD | 92.35% | 90.50% | 91.41% |
| JD | 90.06% | 77% | 83.02% |
| LD | 91.90% | 96.50% | 94.15% |
| MD | 95.58% | 86.50% | 90.81% |
| **PD** | **68.62%** | **82%** | **74.72%** |
| SD | 97.09% | 100% | 98.52% |
| TD | 87.83% | 83% | 85.35% |
| WE | 98.50% | 98.50% | 98.50% |
| Micro Average | | | **88.81%** |

### 4.3 Discussion

The micro-average of the F-score 88.81% is acceptable for text categorization. Note that the domain PD (domestic poli-

tics) receives the lowest F-score. In short, we need a more accurate way of indexing domestic politics. We find that there are many new politicians' names in the news. We collect 1,532 politicians' names and find 68 associated domain events. This forms an initial event structure base. Further experiments using the extracted modifiers and other V-N events are still in progress.

## 5    Conclusion

We define event structure as a meaningful indexing of document. Compare to the traditional indexing, a vector of keywords, an event structure is human readable, since it is a concrete unit of sentence semantics. We apply the domain event structure for document indexing and find that sentences containing a domain event structure can be the summary of a document and can be a good candidate for question answering. Our experiment shows that the result of an automatic ontology acquisition is a good method for text categorization.

## References

[Gruber, 1993] Gruber, T.R., A translation approach to portable ontologies. Knowledge Acquisition, 5(2), pp. 199-220, 1993.

[Hsu et al., 2001] Hsu, W.L., Wu, S.H. and Chen, Y.S., Event Identification Based On The Information Map - InfoMap, in symposium NLPKE of the IEEE SMC Conference, Tucson Arizona, USA.

[Jacquemin & Tzoukermann 1999] Christian Jacquemin and Evelyne Tzoukermann, NLP for Term Variant Extraction: Synergy between Morphology, Lexicon, and Syntax, in Tomek Strzalkowski (ed.) Natural Language Information Retrieval, pp. 25-74, Kluwer Academic Publishers, Netherlands, 1999.

[Smeaton, 1999] Alan F. Smeaton, Using NLP or NLP Resources for Information Retrieval Tasks, in Tomek Strzalkowski (ed.) Natural Language Information Retrieval, pp.99-111, Kluwer Academic Publishers, Netherlands, 1999.

[Strzalkowski *et al.*, 1999] Tomek Strzalkowski, Fang Lin, Jin Wang and Jose Perez-Carballo, Evaluating Natural Language Processing Techniques in Information retrieval, in Tomek Strzalkowski (ed.) Natural Language Information Retrieval, pp.113-145, Kluwer Academic Publishers, Netherlands, 1999.

[Thompson and Dozier, 1999] Paul Thompson and Christopher Dozier, Name Recognition and Retrieval Performance, in Tomek Strzalkowski (ed.) Natural Language Information Retrieval, pp.261-272, Kluwer Academic Publishers, Netherlands, 1999.

[Tsai et al., 2002] Tsai, J. L, Hsu, W.L. and Su, J.W., Word sense disambiguation and sense-based NV event-frame identifier. Computational Linguistics and Chinese Language Processing, 7(1), pp. 1-18.

[Wu et al., 2002] Wu, S.H. and Hsu, W.L., SOAT: A Semi-Automatic Domain Ontology Acquisition Tool from Chinese Corpus, to appear in proceedings of the 19th International Conference on Computational Linguistics (Coling-02), ACM press, 2002.