

Acquiring Selectional Preferences in a Thai Lexical Database

Canasai Kruengkrai, Thatsanee Charoenporn, Virach Sornlertlamvanich, Hitoshi Isahara

Thai Computational Linguistics Laboratory

Communications Research Laboratory

112 Paholyothin Road, Klong 1, Klong Luang, Pathumthani 12120

{canasai, thatsanee, virach}@crl-asia.org, isahara@crl.go.jp

Abstract

In this paper, we consider the problem of enriching a Thai lexical database by extending the semantic information with selectional preferences. We propose a novel approach for acquiring selectional preferences of verbs, which is motivated by the tree cut model. We apply a model selection technique called the Bayesian Information Criterion (BIC). Given a semantic hierarchy, our goal is to generalize initial noun classes to the most plausible levels on that hierarchy. We present an iterative algorithm for generalization. The algorithm performs agglomerative merging on the semantic hierarchy in a bottom-up manner. The BIC is used to measure the improvement of the model both locally and globally. In our experiments, we consider the Web as a large corpus. We also propose approaches for extracting examples from the Web. Preliminary experimental results are given to show the feasibility and effectiveness of our approach.

1 Introduction

For over a decade, researchers in the area of computational linguistics and natural language processing have been interested in the problem of acquiring large semantic knowledge for natural language understanding systems. The availability of lexical databases, such as WordNet (Miller et al., 1993) and

EuroWordNet (Vossen, 1999), appears to be useful for many different research areas, including word sense disambiguation, machine translation, information retrieval, etc. More recently, the reemergence of ontology researches in both theories and applications has activated researchers to reuse and extend these linguistic resources in many other domains.

At the Thai Computational Linguistics Laboratory (TCL), an initial effort has been made to develop a lexical ontology named the *TCL's computational lexicon* by reusing an existing lexical database for machine translation. This lexical database was originally constructed for the use in the Multilingual Machine Translation (MMT) project, which is a six-year (1987-1992) cooperative project among the group of research institutes led by the National Electronics and Computer Technology Center (NECTEC) of Thailand, and the Center of the International Cooperation for Computerization (CICC) of Japan.

The structure of the lexical entry of the TCL's computational lexicon consists of three types of information, including morphological (MOR), syntactic (SYN), and semantic (SEM) information. The morphological information indicates whether the word is a single word or a compound word. The syntactic information gives grammatical categories and subcategories, and verb patterns in sentence structures. The semantic information provides word concepts and case relations. Figure 1 shows an example of word entry ตรวจ 'check'.

Let us focus on the semantic information of verbs. One limitation of the current structure is that thematic roles (or case roles), indicated by MAPS, are

WORD	ตรวจ 3c7e2a CHECK	% Thai word % identifying number linking to description % equivalent English word
MOR	TYPE.{S}	% word formation (single or compound word)
SYN	CAT.{V} SUBCAT.{VACT} VPPAT.{SUB+V+DOB}	% grammatical category of word % grammatical category in more detail % syntactic structure of verb in sentence
SEM	MAPS.{SUB=AGT,DOB=OBJ} AKO.{2-2-6}	% mapping of thematic roles to syntactic structure % a-kind-of relations indicating word position on semantic hierarchy

Figure 1: An example entry of word ตรวจ ‘check’.

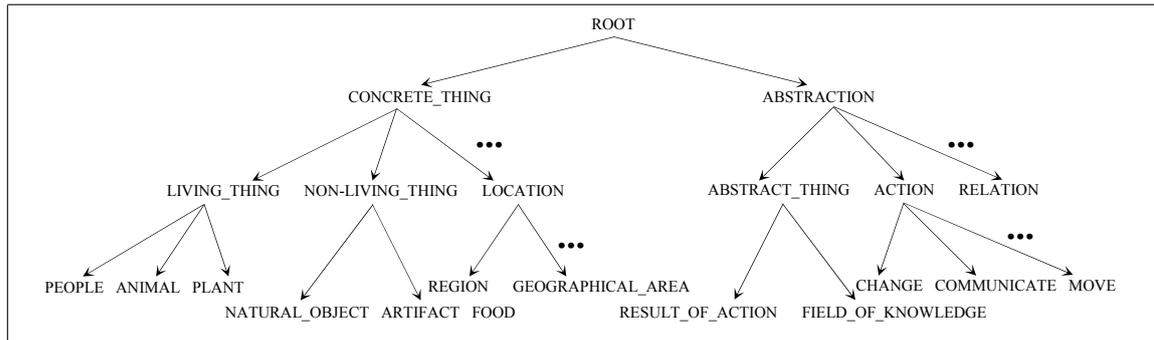


Figure 2: A partition of the semantic hierarchy of the TCL’s computational lexicon.

only mapped to syntactic relationships among words within the same sentence. The idea is to transform the surface structure into the semantic structure. As shown in Figure 1, we know that the subject of the verb ตรวจ ‘check’ is the agent, but we do not know what the semantic class (or concept) of the agent should be. Similarly, we just know that the direct object of the same verb is the object.

To deal with this limitation, we are interested in semantic constraints that are analogous to syntactic constraints called *selectional preferences* or *selectional restrictions* (Manning and Schütze, 1999). For example, the subject of the verb ตรวจ ‘check’ prefers to be humans, the subject of the verb บิน ‘fly’ tends to be birds or airplanes, and the object of the verb ดื่ม ‘drink’ prefers to be beverages. In the TCL’s computational lexicon, each word is mapped onto the semantic hierarchy indicated by AKO. The hierarchy is composed of 189 concept classes. A partition of the semantic hierarchy is shown in Figure 2. Given a verbal predicate, it is a challenging task to find appropriate levels of noun classes on the semantic hierarchy to be selectional preferences.

In this paper, we consider the problem of en-

riching the TCL’s computational lexicon by extending the semantic information with selectional preferences. We focus on a framework for acquiring selectional preferences of verbs. In order to obtain an optimal set of selectional preferences for a given verb, we propose the use of a model selection technique called the Bayesian Information Criteria (Wasserman, 1999). We also propose an efficient algorithm for searching candidate sets and selecting the best one. Preliminarily experimental results are given to show the feasibility and effectiveness of our approach.

We now give a brief outline of the paper. In Section 2, we review related work and give the preamble of the proposed approach. Section 3 describes our approach in detail. In the context of this section, we first explain the BIC and probabilistic model for the semantic hierarchy, and then propose an iterative algorithm for generalization. Section 4 presents the experimental methodology. In Section 5, preliminary results are given with discussion. Finally, Section 6 concludes our work and gives some directions of future work.

2 Related Work

The acquisition of selectional preferences is the operation of finding suitable classes on a semantic hierarchy for predicates. Most algorithms for selectional preference induction are based on corpus-based approaches. The process can be broadly classified into three steps (Ribas, 1995). The steps are to create the space of candidate classes from examples, evaluate the appropriateness of the candidates using some statistical measures, and select the most optimal candidates to stand for the selectional preferences.

Resnik (1993) proposed a class-based model that utilizes information theory and statistical modeling. Based on deriving a semantic hierarchy from WordNet, the approach first calculates association scores of all candidate noun classes for a given verb. It then selects the noun class having the maximum association score. Ribas (1995) addressed some shortcomings of Resnik’s approach. Several improving techniques were proposed, such as modifying statistical measures to deal with noise in the data, and using word sense disambiguation to scope the space of noun classes.

The tree cut model was proposed by Li and Abe (1998). The approach also reuses WordNet as the semantic hierarchy. It estimates conditional probability distributions over possible partitions of nouns using the maximum likelihood estimate, and selects the best partition through the Minimum Description Length (MDL) principle (Rissanen and Ristad, 1994). McCarthy (2000) also applied the tree cut model to the problem of identifying diathesis alternations. Wagner (2000) proposed a variation of the tree cut model by introducing a weighting factor to the log-likelihood of the data. Recently, other statistical approaches, such as the Bayesian Networks (Ciaramita and Johnson, 2000) and the hidden Markov models (HMM) (Abney and Light, 1999), have been investigated.

This paper presents a novel approach for selectional preference acquisition, which is motivated by the tree cut model. We apply a model selection technique called the Bayesian Information Criterion (BIC) for obtaining an optimal model. In our case, we need to find a set of noun classes to be selectional preferences for a given verb. We can consider this problem as model selection. Fortunately, we in-

herently have the semantic hierarchy from the core structure of the TCL’s computational lexicon. Our goal is to generalize initial noun classes to the most plausible levels on the semantic hierarchy. We propose an iterative algorithm that performs agglomerative merging on the hierarchy in a bottom-up manner. The BIC is used to measure the improvement of the model both locally and globally. In our experiments, we consider the Web as a large corpus. We also propose approaches for extracting examples from the Web.

3 Selectional Preference Acquisition

3.1 Bayesian Information Criterion for Semantic Hierarchy

The Bayesian Information Criterion (BIC) is one of techniques for model selection (Wasserman, 1999). The problem of model selection is to choose the best one among a set of candidate models $\mathbf{m}_i \in \mathcal{M}$. The BIC of a model \mathbf{m}_i can be approximated as follows:

$$BIC(\mathbf{m}_i) = \hat{l}_i(D) - \frac{p_i}{2} \cdot \log|D|, \quad (1)$$

where $\hat{l}_i(D)$ is the log-likelihood of the data D according to \mathbf{m}_i , and p_i is the number of independent parameters. The BIC has several interesting characteristics (Chickering and Heckerman, 1997). On the one hand, it is independent of the prior. On the other hand, it is exactly minus the MDL.

We adopt the tree cut model to characterize the probabilistic model of the semantic hierarchy. Let $\mathbf{m} = (\Gamma, \Theta)$ be the model, including a partition in the semantic hierarchy being considered Γ , and parameters Θ . Given the noun class $C \in \Gamma$, the verb $v \in \mathcal{V}$, and the syntactic relationship $r \in \mathcal{R}$, the conditional probability distribution of $\hat{P}(C|v, r)$ must satisfy:

$$\sum_{C \in \Gamma} \hat{P}(C|v, r) = 1. \quad (2)$$

There are two important assumptions for estimating probabilities in this model. First, for any noun $n \in C$, the probability can be estimated by using the maximum likelihood estimate (MLE). Second, for any class C , the probability is distributed uniformly to all nouns dominated by it. Based on these assumptions, the probability of the noun n can be

calculated by:

$$\hat{P}(n) = \frac{\hat{P}(C)}{|C|}, \quad (3)$$

and

$$\hat{P}(C) = \frac{\sum_{n \in C} \text{freq}(n)}{|D|}, \quad (4)$$

where $\text{freq}(n)$ is the frequency of the noun n co-occurring with the verb v and the syntactic relationship r , $|D|$ is the size of the data (or the total frequency of all nouns), and $|C|$ is the number of classes in the current partition. Thus, the log-likelihood of class C according to \mathbf{m}_i is:

$$\hat{l}_i(C) = \log \prod_{n \in C} \hat{P}(n) = \sum_{n \in C} \log \hat{P}(n). \quad (5)$$

From Equation 1, we can write:

$$BIC(\mathbf{m}_i) = \sum_{C \in \Gamma} \hat{l}_i(C) - \frac{p_i}{2} \cdot \log |D|, \quad (6)$$

where the number of parameters p_i is equivalent to the number of classes in Γ minus one, $|C| - 1$. Finally, we can write the following objective function:

$$\mathbf{m}^* = \operatorname{argmax}_{\mathbf{m}_i \in \mathcal{M}} BIC(\mathbf{m}_i). \quad (7)$$

3.2 The Agglomerative Merging Algorithm for Generalization

We now describe an iterative algorithm for selectional preference generalization. Our algorithm searches the appropriate levels of noun classes on the semantic hierarchy by performing agglomerative merging in a bottom-up manner. One may think of the behavior of the algorithm as a simplified agglomerative clustering algorithm. We assume that all nouns are pre-classified onto their hierarchical classes according to the semantic information indicated by AKO. As a result, the algorithm does not have to make any decision about assigning nouns to the most probable classes. What it has to do is to repeatedly merge subclasses into a single class if the structure of the semantic hierarchy improves. We consider this structure as a model for representing selectional preferences. The improvement of the model can be measured by using the BIC as described in the previous section. The more the BIC

Algorithm 1: MERGECLASSES($\{c_i\}_{i=1}^k$)

```

begin
   $c' \leftarrow \emptyset$ ;
  for  $i$  from  $i = 1, \dots, k$  do
     $c' \leftarrow c' \cup c_i$ ;
  endFor
  if  $BIC(c') > BIC(\{c_i\}_{i=1}^k)$  then
    return  $c'$ ;
  else
    return  $\emptyset$ ;
  endif
end

```

Algorithm 2: AGGLOMERATIVEMERGING

```

input : Semantic hierarchy  $\Gamma$  containing a
        set of initial leaf nodes  $c_i$ , where
         $i = 1, \dots, m$ .
output : Generalized  $\Gamma$  with leaf nodes
        forming the optimal noun classes.

begin
  repeat
    Find remaining nodes to merge,
     $\{c_i\}_{i=1}^k$ ;
    if  $k = 0$  then
      break;
    endif
     $c' = \text{MERGECLASSES}(\{c_i\}_{i=1}^k)$ ;
    if  $c' \neq \emptyset$  then
       $\Upsilon = \Gamma \setminus \{c_i\}_{i=1}^k \cup c'$ ;
      if  $BIC(\Upsilon) > BIC(\Gamma)$  then
        Re-distribute  $\hat{P}(n)$  for  $n \in c'$ 
        according to Equation 3;
        DELETE( $\Gamma, \{c_i\}_{i=1}^k$ );
        APPEND( $\Gamma, c'$ );
         $m = m - k + 1$ ;
      endif
    endif
  until  $m < 1$ ;
end

```

increases, the more the model improves. The agglomerative merging algorithm tries to increase the objective function value in Equation 7 at every step. Thus, the BIC is used to test the improvement of the model both locally and globally.

Our algorithm proceeds as following. It starts by

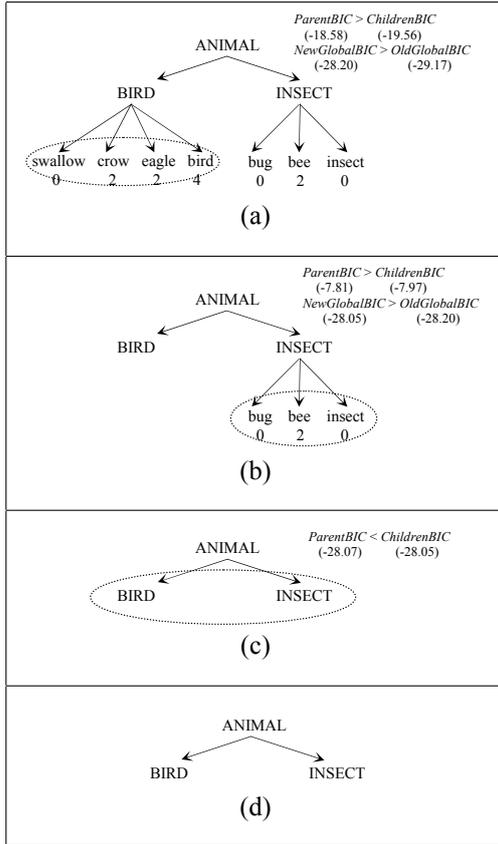


Figure 3: Agglomerative merging of a simple semantic hierarchy.

initializing the region of noun classes on the semantic hierarchy Γ . The input data are given in the form of co-occurrence tuple, $\langle v, r, n, freq \rangle$, where v is the verb, r is the syntactic relationship, n is the noun, and $freq$ is the co-occurring frequency. The approaches for obtaining these data are described in Section 4.1 and 4.2. It then finds appropriate leaf nodes having the same AKO to merge up into the parent node. Focusing on this partition, the BIC is measured locally. If the BIC score of the parent node is not greater than the BIC score of the children nodes, the algorithm keeps the structure of leaf nodes as it is. Otherwise, the BIC is measured globally to guarantee the overall improvement. These processes are given in Algorithm 1 and 2. The algorithm iterates until it cannot find leaf nodes to merge or there remains one class.

Figure 3 illustrates an example of how the algorithm works, which is reproduced from (Li and Abe, 1998). Given the verb *fly* with the syntactic relation-

ship *subject*, the co-occurring nouns are: crow (2), eagle (2), bird (4), and bee (2), where numbers in the parentheses indicate the co-occurring frequency of nouns. Let us focus on Figure 3a, which is the initial semantic hierarchy of the data. The algorithm starts by finding possible leaf nodes to merge. Since the local BIC score increases, it further measures the global BIC score by comparing the overall structure. The global BIC score also increases, it decides to merge the children nodes into the parent node. Figure 3b performs the same process. In Figure 3c, since the local BIC score decreases, it is not necessary to measure the global BIC score. Finally, we obtain the generalized semantic hierarchy in 3d, whose remaining leaf nodes are considered to be selectional preferences.

4 Experimental Methodology

4.1 Collecting Data from the Web

As mentioned earlier, we view the Web as a large and free corpus. Below we describe how to retrieve examples for selectional preference generalization through search engines. Common search engines usually return results, including a number of relevant links and their short descriptions. Since our objective is to extract the co-occurrence tuples, what we anticipate from the search engines is that, given a verb as a query, the returned short descriptions may contain the verb and its context. We refer to these short descriptions as *snippets*.

We implemented a simple web robot that sends the target verb to the search engines, and retrieves all the search results kept into a repository. Two major search engines of Thailand were used, including www.sansarn.com and www.siamguru.com. Then, we parsed HTML documents in the repository to extract only snippets. We obtained about 800-1000 snippets for each verb query. Each snippet contains 100-150 words on average. Figure 4 shows snippets of the verb ตรวจ ‘check’ extracted from search results.

The benefits of using the snippets from the search engines are two folds. On the one hand, we can use the efficient search mechanism to get the context of the target word without implementing any string-pattern matching algorithms. On the other hand, we obtain the large databases of the search engines,

... รอง สว.ผ.2 กก.2 ปล. ได้ร่วมประชุมเพื่อ**ตรวจ**สำนวนและเอกสารหลักฐานที่เกี่ยวข้องกับคดีเพื่อที่จะหาทางดำเนินคดี ...
 ... สอแววยืดเขื่อนไม่รู้จักบังคับใช้ได้เมื่อไหร่ ด้าน อุทัย ขอเวลา 15 วัน **ตรวจสอบ**รายชื่อผู้สนับสนุนร่างกฎหมาย ...
 ... แพทย์ประจำตัวหลวงพ่อดุณ และแนะนำให้นำ**ตรวจ**ร่างกายอย่างละเอียด หลังจากเข้าพักรักษาตัว ...
 ... ล่าสุดตำรวจป่าไม้ได้เข้า**ตรวจ**ค้นโรงงานเชือดสัตว์ป่าอีกแห่งที่กรุงเทพฯ พบลูกสิงอุรังอุตั้งแข่งสภาพนำสงสาร ...
 ... ตามจุดต่างๆ รอบๆ บริเวณโรงแรม ก็มีเจ้าหน้าที่ตำรวจตั้งด่าน**ตรวจ**รถทุกคันที่จะผ่านเข้าออก ส่วนที่ชั้นล่างของโรงแรม ...

Figure 4: Snippets of the verb ตรวจ ‘check’ extracted from search results.

reflecting natural language usage in the society.

One problem we faced is that the snippets are too heterogenous. For example, since the descriptions of the web pages were produced from table data containing lists of items or bullets, the snippets did not contain grammatical features and were less meaningful. Consequently, we limited our web robot to crawl particularly on news sites, which are already categorized by both search engines. The search results from the news categories seem to contain more useful phrases having the target verb with its context.

4.2 Extracting Co-occurrence Tuples

Since we need the final input data of the algorithm in the form of the co-occurrence tuple, $\langle v, r, n, freq \rangle$, linguistic tools for analyzing morphological and syntactic structure of Thai text are required. However, we only have a parts-of-speech tagger called Swath.¹ A syntactic relationship between a target verb v and its co-occurring noun n is manually assigned. In this section, we describe an approach that assists human subjects to do such task.

After retrieving snippets containing the target verb v and its context, we do word segmentation and parts-of-speech tagging by using Swath. Note that Thai text has no explicit word boundaries like English text, so we have to segment it into meaningful tokens. We consider ± 3 words of context around the target verb v . This window size is enough to capture syntactic relationships. Now we can think that we have the tuple structure like $\langle v, context\ relationship, n, freq \rangle$. Thus, we need to transform a context relationship to an appropriate syntactic relationship r .

We observe that the co-occurring frequencies have small different values. In order to filter out

¹The software is publicly available at <http://www.links.nectec.or.th/~yai/software.html>.

Word	$Freq$	LLR_{+3}
ร่างกาย ‘body’	9	24.6864
หนังสือเดินทาง ‘passport’	2	6.4391
กล้ามเนื้อ ‘muscle’	1	4.5825
พยาธิ ‘worm’	1	4.5825
ป่าไม้ ‘forest’	2	4.3856
คฤหาสน์ ‘mansion’	2	4.3856
รถบรรทุก ‘truck’	2	3.7537
บ้านพัก ‘home’	2	2.8196
กระเป๋ ‘bag’	2	2.8196
สุขภาพ ‘health’	1	2.6056
ผลิตภัณฑ์ ‘product’	1	1.4067
รถ ‘car’	8	1.3848
โรงงาน ‘factory’	1	1.0653
กะโหลก ‘skull’	2	0.9742

Table 1: Co-occurring nouns of verb ตรวจ ‘check’ within window size +3.

nouns that have insignificant dependence of the target verb, we measure dependence between words by using statistics taken from all the snippets. We apply the log likelihood ratio (LLR) (Dunning, 1994) for selecting the most optimal nouns. Given the verb v and the noun n occurring within window size z , a fast version of the LLR can be calculated as follows (Tanaka, 2002):

$$LLR_z(v, n) = k_{11} \log \frac{k_{11}N}{Q_1 R_1} + k_{12} \log \frac{k_{12}N}{Q_1 R_2} + k_{21} \log \frac{k_{21}N}{Q_2 R_1} + k_{22} \log \frac{k_{22}N}{Q_2 R_2}, \quad (8)$$

$$k_{11} = freq(v, n),$$

$$k_{12} = freq(v) - k_{11},$$

$$k_{21} = freq(n) - k_{11},$$

$$k_{22} = N - k_{11} - k_{12} - k_{21},$$

$$Q_1 = k_{11} + k_{12}, Q_2 = k_{21} + k_{22},$$

$$R_1 = k_{11} + k_{21}, R_2 = k_{12} + k_{22},$$

where $freq(v, n)$ is the co-occurring frequency between v and n , $freq(v)$ and $freq(n)$ are frequencies of v and n , respectively.

We were left only nouns with their LLR values above a pre-defined threshold. Table 1 shows the top 14 co-occurring nouns within window size +3 for a given verb ตรวจ ‘check’. The second and third columns show their co-occurring frequencies and LLR values, respectively. The nouns within the window size -3 are considered in the similar way. Once the candidate nouns are produced, we ask human subjects to analyze and assign the most suitable syntactic relationships between the verb and candidate nouns. For example, from Table 1, we get co-occurrence tuples (ตรวจ ‘check’, obj, ร่างกาย ‘body’, 9), (ตรวจ ‘check’, obj, หนังสือเดินทาง ‘passport’, 2), and so on.

5 Results and Discussion

Evaluating selectional preference generalization is a difficult task, because it requires the *gold standard* results for making comparisons. Those gold standard results may be produced by using the majority of the human agreements. At the present, we have no such gold standard for Thai language. However, in order to observe the behavior of our algorithm, we selected Thai verbs, including ตรวจ ‘check’, สร้าง ‘build’, ซื้อ ‘buy’, and จ่าย ‘pay’ for evaluation. We considered two syntactic structures, including subject-verb and verb-direct object relationships. Table 2 and 3 show examples of generalization results that seem to be close to human intuition. For example, the subject of the verb ตรวจ ‘check’ falls into the class PEOPLE, which its children classes are PERSON and ORGANIZATION. The class ANIMAL_PART can be discovered to be the object of the same verb. The computational time is very short, which is less than one second running on a personal computer with Pentium processor 2GHz and memory 512 KB.

In addition, we observe that the noun sense ambiguity can lead to irrelevant results in some cases. For example, the noun โรงพยาบาล ‘hospital’ has two senses, which are categorized into two classes: CONSTRUCTION and ORGANIZATION. However, the class CONSTRUCTION is unlikely to be the subject of the verb ตรวจ ‘check’. Since the tree cut model just deals

Class	Prob.	Word Example
Subject of ตรวจ ‘check’		
PEOPLE	1.00	ตำรวจ ‘police’
Subject of สร้าง ‘build’		
ABSTRACT_THING	0.69	สังคม ‘society’
ORGANIZATION	0.04	รัฐบาล ‘government’
PERSON	0.03	นักท่องเที่ยว ‘tourist’
Subject of ซื้อ ‘buy’		
PERSON	0.40	ชาวบ้าน ‘villager’
CONSTRUCTION	0.35	โรงพยาบาล ‘hospital’
ORGANIZATION	0.25	บริษัท ‘company’
Subject of จ่าย ‘pay’		
PERSON	0.54	นักเรียน ‘student’
CONSTRUCTION	0.39	ธนาคาร ‘bank’
CULTURAL_ABSTRACT_THING	0.08	ประธาน ‘chairman’

Table 2: Generalization results with subject-verb relationship.

Class	Prob.	Word Example
Direct Object of ตรวจ ‘check’		
ARTIFACT	0.34	รางวัล ‘prize’
ABSTRACT_THING	0.22	เอกสาร ‘document’
ANIMAL_PART	0.18	ร่างกาย ‘body’
Direct Object of สร้าง ‘build’		
ABSTRACT_THING	0.65	มาตรการ ‘measure’
ARTIFACT	0.16	สะพาน ‘bridge’
ATTRIBUTE	0.10	สถานการณ์ ‘situation’
Direct Object of ซื้อ ‘buy’		
ABSTRACT_THING	0.40	ธุรกิจ ‘business’
ARTIFACT	0.27	ของที่ระลึก ‘souvenir’
GRAIN	0.03	ข้าว ‘rice’
Direct Object of จ่าย ‘pay’		
IMMATERIAL_THING	0.31	ค่าเช่า ‘rental fee’
SOCIAL_ABSTRACT_THING	0.22	ค่า ‘fee’
RESULT_OF_ACTION	0.01	ดอกเบี้ย ‘interest’

Table 3: Generalization results with verb-direct object relationship.

with this problem by equally dividing the frequency of a noun among all the classes containing that noun, more sophisticated approach is needed for further improvement of our algorithm.

6 Conclusion and Future Work

In this paper, the problem of enriching the TCL’s computational lexicon has been considered. We present approaches for acquiring selectional preferences. We apply the BIC for selecting the optimal model. We propose an agglomerative merging algorithm, which is capable of generalizing noun classes on the semantic hierarchy. The preliminary results

are very encouraging.

In future work, we plan to pursue the following issues. In preprocessing, a parser that can analyze the syntactic structure of text will be developed. This can help to automatically produce the input data of the algorithm in the form of co-occurrence tuples without human participants. For the algorithm, several methods for solving noun sense ambiguity will be investigated (Yarowsky, 1995) (Schütze, 1998). In (Abe and Li, 1996), the authors show that combining the association norm with the MLE can improve the accuracy of generalization. We believe that it can be effectively applied to our algorithm.

Acknowledgements

The authors would like to thank Prapass Srivastava for providing us an initial version of the web robot program, and the anonymous reviewers for their valuable comments.

References

- Abe, N., and Li, H. 1996. Learning word association norms using tree cut pair models. In *Proceedings of the Thirteenth International Conference on Machine Learning*.
- Abney, S., and Light, M. 1999. Hiding a semantic hierarchy in a markov model. In *Proceedings of the ACL Workshop on Unsupervised Learning in Natural Language Processing*.
- Chickering, D. M., and Heckerman, D. 1997. Efficient approximations for the marginal likelihood of bayesian networks with hidden variables. *Machine Learning*, 29, pages 181–212.
- Ciaramita, M., and Johnson, M. 2000. Explaining away ambiguity: Learning verb selectional preference with bayesian networks. In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 187–193.
- Dunning, T. 1994. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1): 61–74.
- Li, H., and Abe, N. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2): 217–244.
- Manning, C., and Schütze, H. 1999. *Foundations of statistical natural language processing*. MIT Press. Cambridge, MA.
- McCarthy, D. 2000. Using semantic preferences to identify verbal participation in role switching alternations. In *Proceedings of the first Conference of the North American Chapter of the Association for Computational Linguistics*.
- Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K. 1993. *Introduction to WordNet: An on-line lexical database*. CSL Report 43.
- Resnik, P. 1993. *Selection and information: A class-based approach to lexical relationships*. Doctoral Dissertation, Department of Computer and Information Science, University of Pennsylvania.
- Ribas, F. 1995. *On acquiring appropriate selectional restrictions from corpora using a semantic taxonomy*. Ph.D. thesis, University of Catalonia.
- Ribas, F. 1995. On learning more appropriate selectional restrictions. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*.
- Rissanen, J., and Ristad, E. 1994. Language acquisition in the mdl framework. In Eric Sven Ristad, *Language Computation*. American Mathematical Society, Philadelphia.
- Schütze, H. 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1): 97–123.
- Tanaka, T. 2002. Measuring the similarity between compound nouns in different languages using non-parallel corpora. In *Proceedings of the 19th International Conference on Computational Linguistics*.
- Vossen, P. 1999. *EuroWordNet general document*. EuroWordNet (LE2-4003,LE4-8328).
- Wagner, A. 2000. Enriching a lexical semantic net with selectional preferences by means of statistical corpus analysis. In *Proceedings of the ECAI-2000 Workshop on Ontology Learning*, pages 37–42.
- Wasserman, L. 1999. Bayesian model selection and model averaging. *Journal of Mathematical Psychology*.
- Yarowsky, D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. Cambridge, MA, pages 189–196.