A Constraint-Based Approach for Computational Lexicon Construction

Thatsanee Charoenporn[1,2], Canasai Kruengkrai[1], Virach Sornlertlamvanich[1],
Thanaruk Theeramungkong[2], and Hitoshi Isahara[1]

[1]Thai Computational Linguistics Laboratory
National Institute of Information and Communication Technology
112 Paholyothin Road, Klong 1, Klong Luang, Pathumthani 12120, Thailand
{thatsanee,canasai,virach}@tcllab.org, isahara@crl.go.jp
[2]Sirindhorn International Institute of Technology
Pathumthani 12120, Thailand
ping@siit.tu.ac.th

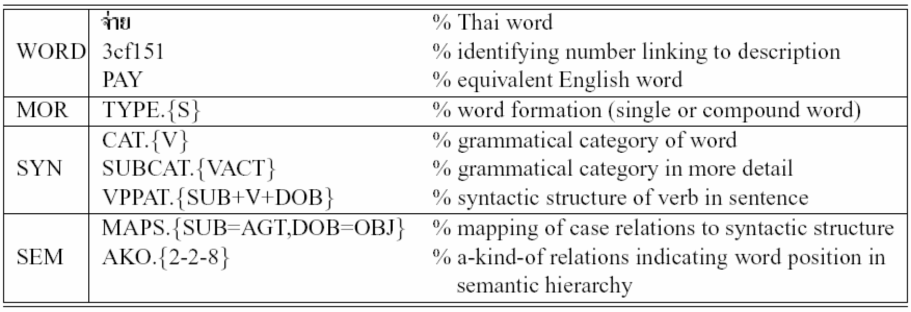
Abstract. Ontologies are the central component for the Semantic Web, since they can be used to explicitly represent the semantics of structured or semi-structured information. In this paper, we describe the recent developments of a lexical ontology named the TCL's computational lexicon, which aims to serve as the core knowledge base for the Semantic Web. We focus on designing a new specification of the semantic information based on the logical and semantic constraints. We sketch practical approaches to acquire these constraints by reusing and extending existing linguistic resources.

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 and EuroWordNet, 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.

Although English is used as the major language that mostly dominates web content, fragments of many other languages constantly increase. At the Thai Computational Linguistics Laboratory (TCL), an initial effort has been made to develop a lexical ontology named the TCL's computational lexicon. The TCL's computational lexicon aims to serve as the core knowledge base for NLP research communities and real-world applications, e.g. the Semantic Web.

The current structure of the lexical entry of the TCL's computational lexicon composes of three types of information: morphological (MOR), syntactic (SYN), and semantic (SEM) information. Figure 1 shows the lexical entry of the verb จ่าย 'pay'. Let us focus on the semantic information. It contains two slots, including MAPS and AKO. The MAPS only links thematic roles (case roles) to syntactic relationships among the circumstance words within the same sentence. The AKO indicates the word position in the semantic hierarchy. However, using these types of the semantic information is inadequate for representing and discriminating word meanings, particularly on the lack of other semantic relations.



WORD	จ่าย 3cf151 PAY	% 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.{2-2-8}
		% mapping of case relations to syntactic structure
		% a-kind-of relations indicating word position in semantic hierarchy

Figure 1: An example entry of the verb จ่าย 'pay'

In this paper, we describe the recent developments in reusing existing linguistic resources for enriching the TCL's computational lexicon. We focus on designing a new specification of the semantic information based on the logical and semantic constraints. We also consider how to acquire the semantic information in automatic and semi-automatic ways rather than using only manual annotation. We sketch practical approaches to perform such tasks. A by-product of our new semantic structure is that we derive a methodology of word sense representation in contrast to the descriptive manner using in general lexical databases.

The remainder of the paper is organized as follows: In Section 2, we briefly describe the background of the TCL's computational lexicon. Section 3 presents our proposal of the new structured form of the semantic information. In Section 4, we describe how to generalize semantic constraints to appropriate noun classes. Finally, Section 5 concludes our paper with some directions of future work.

2 Background of the TCL's Computational Lexicon

The TCL's computational lexicon is developed by reusing an existing lexical database for machine translation. It contains 69,060 lexical entries of Thai words. This lexical database was originally constructed for using in the Multilingual Machine Translation (MMT) project, which is a six-year (1987-1992) cooperative project between groups 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, syntactic, and semantic information. The morphological information indicates types of word composition (TYPE). The syntactic information gives grammatical categories (CAT) and subcategories (SUBCAT), and verb patterns in sentence structures (VPPAT). The semantic information provides word concepts (AKO) and case relations (MAPS).

The types of word composition encoded in the morphological information are classified into 2 types; single word as โทรศัพท์ 'telephone' and compound word as โทรศัพท์มือถือ 'mobile phone'. A single word is a lexical unit composed of either monosyllable or polysyllable and referring to only one conceptual meaning. Contradictorily, a compound word is composed of more than one single word, e.g. the word โทรศัพท์มือถือ 'mobile phone' consists of 2 lexical units, โทรศัพท์ 'telephone', and มือถือ 'mobile'.

The syntactic information contains all the information on the syntactic structure of each word, including grammatical categories and subcategories, and the pattern of the verb in a grammatical sentence structure. The TCL's computational lexicon carries 11 categories with 44 subcategories for narrowing down the functions of words. Each category is divided into subcategories by their position, composition, and reference. For example, the category "determiner" is divided into 9 subcategories by its occurring position, such as after noun, after noun and classifier, between noun and classifier, etc.

For the verb pattern, the lexicon classifies the pattern of the verb according to the position of the obligatory arguments and its contextual environment into 11 patterns. For example, the pattern "SUB+V+DOB" indicates that the syntactic structure of this verb consists of the other two components: a NP as a subject (SUB) and a NP as a direct object (DOB).

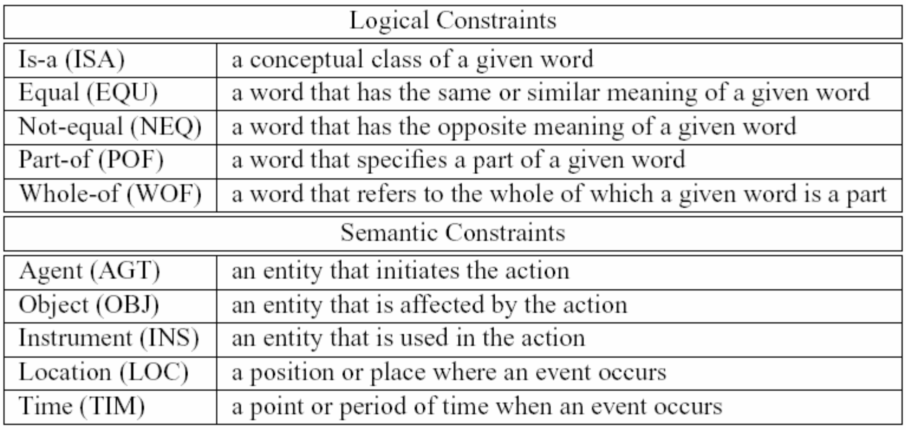
For the semantic information, the TCL's computational lexicon gives basic information about case relations and word concepts. The case relations just bind the thematic roles with syntactic arguments, normally between the main verb and its pre- and post-components. The word concepts are organized on the semantic hierarchy. Each word concept is a group of lexical entries classified and ordered in a hierarchical level of meanings. The semantic hierarchy is composed of 189 concept classes.

3 Acquiring Semantic Information

3.1 Forming Semantic Structure with Constraints

Our task is to redefine the structured form of the semantic information in the TCL's computational lexicon. As addressed previously, the current structure is not expressive to all possible semantic representations. The thoroughness of the representation is required. Consequently, we propose to solve those shortcomings by forming the semantic structure with logical and semantic constraints. The main idea of our design is based on the simplicity and re-usability. The logical constraints are capable of dealing with the absence of relatedness of word meanings. The semantic constraints try to discover preferences of syntactic arguments of thematic roles.

Table 1 shows the entire constraints with their descriptions. One can observe that some logical constraints are similar to semantic relations used to construct WordNet. The EQU, NEQ, POF, and WOF constraints are equivalent to synonyms, antonyms, meronyms and holonyms, respectively. For the ISA constraint, it is identical to AKO of the word. The semantic constraints are the same as the thematic roles, but they are specified with selectional preferences. Here we limit the thematic roles into five important roles, including AGT, OBJ, INS, LOC, and TIM.



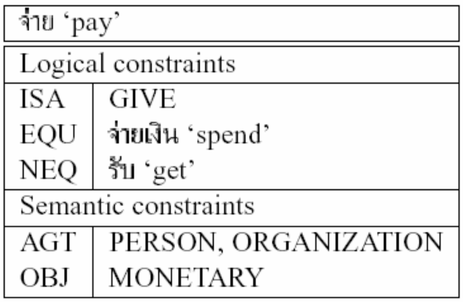
Logical Constraints	
Is-a (ISA)	a conceptual class of a given word
Equal (EQU)	a word that has the same or similar meaning of a given word
Not-equal (NEQ)	a word that has the opposite meaning of a given word
Part-of (POF)	a word that specifies a part of a given word
Whole-of (WOF)	a word that refers to the whole of which a given word is a part
Semantic Constraints	
Agent (AGT)	an entity that initiates the action
Object (OBJ)	an entity that is affected by the action
Instrument (INS)	an entity that is used in the action
Location (LOC)	a position or place where an event occurs
Time (TIM)	a point or period of time when an event occurs

Table 1 : Logical and Semantic Constraints for the semantic information.

It is not necessary that a lexical item must be encoded with all the constraints. We can see that the logical constraints capture information of relations among nouns, while the semantic constraints connect verbs and related nouns. Furthermore, we classify the constraints into two types: obligatory and optional. While the obligatory constraints should be filled as much as possible, the optional constraints can be left empty.

In the logical constraints, there is only one obligatory constraint, ISA. It is also considered to be the core structure of the TCL's computational lexicon. In the semantic constraints, AGT and OBJ are the obligatory constraints. The remaining constraints are categorized to be the optional constraints.

Based on the idea of the frame representation, we finally consider our semantic structure as a template that provides slots of attributes and values to compose the meaning of a given word. For example, the new semantic structure of the verb จ่าย 'pay' is shown below.



จ่าย 'pay'	
Logical constraints	
ISA	GIVE
EQU	จ่ายเงิน 'spend'
NEQ	รับ 'get'
Semantic constraints	
AGT	PERSON, ORGANIZATION
OBJ	MONETARY

The problem is how to fill the values, which will be described in the next section.

3.2 Acquisition Schemes

We now describe how to acquire the information of the semantic structure, which is composed of the logical and semantic constraints. The main idea of our approach relies on reusing existing linguistic resources and annotating the structure in automatic and semi-automatic ways.

3.2.1 Logical Constraint Acquisition

The logical constraints can be acquired by extracting information from the TCL's computational lexicon itself and the other Thai lexical database named LEXiTRON (http://lexitron.nectec.or.th), which is an online Thai-English dictionary. Since the current semantic structure already has AKO that indicates the same information as what we need for ISA, we can use it directly. Rather than filling the value of this constraint with the identifying number of the position on the semantic hierarchy, we put the name of the concept explicitly.

EQU and NEQ can be obtained from corresponding lexical entries in. It contains two relations among words: synonyms and antonyms. As described earlier, EQU and NEQ can be interpreted as synonyms and antonyms. Therefore, we can automatically map these relations to fill the values in EQU and NEQ.

Acquiring POF and WOF is a more difficult task. These two components are analogous to meronyms and holonyms. At the present, we do not fill the values of these components. However, it is possible to automatically obtain these kinds of information. Recently, several approaches have been proposed. Berland and Charniak [1] provide a mechanism for extracting meronyms by finding lexical patterns that tend to indicate part-whole relations. Sundblad [9] describes the use of question corpora for acquiring meronyms. The idea is similar to Hearst [3] which performs discovering hyponyms from unrestricted text corpora based on lexico-syntactic patterns. Girju et al. [2] propose an alternative approach for finding meronyms and holonyms based a supervised learning algorithm.

3.2.2 Semantic Constraint Acquisition

The semantic constraints can be acquired by identifying selectional preferences of verbal predicates. Here we focus on the obligatory constraints, including AGT and OBJ. As shown in Figure 1, we know that the subject of the verb จ่าย 'pay' is the agent, but we do not know what the semantic class (concept) of the agent should be. Typically, one may think that the subject of the verb จ่าย 'pay' prefers to be humans. By parsing through text corpora, we can obtain examples of context nouns that are considered to be the subjects of the verb.

In our study, we view the Web as large and free corpora. We can retrieve useful data through search engines. Common search engines usually return results, including a number of relevant links with their short descriptions. Since our objective is to extract context nouns of verbs for analyzing syntactic relationships, 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 implement 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 are used, including www.sansarn.com and www.siamguru.com. We parse HTML documents in the repository to extract only snippets. We obtain about 800-1000 snippets for each verb query. Each snippet contains 100-150 words on average.

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, reflecting natural language usage in the society.

However, it is too fine-grained to tag the semantic constraints with all the corresponding nouns. The problem is how to generalize these nouns to the semantic classes. It is a challenging task to find appropriate levels of noun classes on the semantic hierarchy to be selectional preferences. The approach for dealing with this problem is given in Section 4.

3.3 Word Sense Representation

As mentioned earlier, we derive a methodology of word sense representation from our semantic structure. In stead of describing each word sense with natural language as in general dictionaries, our lexical items are encoded with the logical and semantic constraints. In this section, we roughly describe the concept of the methodology.

We adapt the notation from Kifer et al. [4]. The structure of the representation is in the form:

$$h\ [\ c_1 \rightarrow \{v_{11}, v_{12}, \ldots\}; c_2 \rightarrow \{v_{21}, v_{22}, \ldots\}; \ldots\ ]$$

where h is a head word taken from an equivalent English word, is a type of the logical or semantic constraints as shown in Table 1, v_{ij} is a value of the constraints that can be a concept or semantic class. C is a set of concepts and semantic classes in the lexicon.

The logical and semantic constraints can help to identify and discriminate word senses. For example, the verb จ่าย 'pay' has two senses, including the sense of "to pay some money" and "to distribute something". By using the logical and semantic constraints from the semantic information, we can write:

pay	[ISA → {GIVE};	**distribute**	[ISA → {GIVE};
	EQU → {จ่ายเงิน 'spend'};		EQU → {แบ่ง 'divide'};
	NEQ → {รับ 'get'};		AGT → {MACHINERY,
	AGT → {PERSON,		PERSON,
	ORGANIZATION};		ORGANIZATION};
	OBJ → {MONETARY}]		OBJ → {ARTIFACT}]

We can see that both senses can be systematically distinguished with the values of the constraints. Further studies are required for optimizing word sense representation.

These include computational operations among word senses, such as addition, deletion, comparison, and generalization.

4 Generalizing Semantic Constraints

4.1 Model Selection on Semantic Hierarchy

In order to tag the semantic constraints with the semantic classes, we present an approach for selectional preference acquisition, which is motivated by the tree cut model. The tree cut model was proposed by Li and Abe [6]. The approach 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) principal [8]. McCarthy [7] also applied the tree cut model to the problem of identifying diathesis alternations. Wagner [10] proposed a variation of the tree cut model by introducing a weighting factor to the log-likelihood of the data.

Here we propose the use of a model selection technique called the Bayesian Information Criteria (BIC) for obtaining an optimal model. The BIC has several interesting characteristics. On the one hand, it is independent of the prior. On the other hand, it is exactly minus the MDL. In our case, we need to find a set of noun classes to be selectional preferences for a given verb. Fortunately, we inherently have the semantic hierarchy from the core structure of the TCL's computational lexicon. We try to generalize initial noun classes to appropriate levels on the semantic hierarchy. This problem can be considered as model selection. We apply the BIC to measure the improvement of the model.

4.2 Selectional Preference Generalization

In this section, we 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. We can 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. The more the BIC increases, the more the model improves. The agglomerative merging algorithm tries to increase the overall BIC score at every step. Thus, the BIC is used to test the improvement of the model both locally and globally.

Our algorithm starts by initializing the region of noun classes on the semantic hierarchy. The input data are given in the form of the co-occurrence tuple, <*v, r, n,* *freq*>, where *v* is the verb, *r* is the syntactic relationship, *n* is the noun, and *freq* is the co-occurring frequency. The co-occurrence tuples can be obtained by extracting and analyzing the snippets. 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. The algorithm iterates until it cannot find leaf nodes to merge or there remains one class. More details of our algorithm can be found in Kruengkrai et al. [5].

5 Conclusion and Future Work

In this paper, we have described the ongoing work on developing the TCL's computational lexicon. We redefine the structured form of the semantic information by using the logical and semantic constraints. Our specification attempts to cover all possible semantic representation based on the concept of the simplicity and re-usability. The acquisition schemes of the semantic information are given.

In future work, we plan to explore approaches to extract other logical and semantic constraints. Additionally, we further study how the computational operations among word senses can be performed based on our representation proposed in this paper.

References

1. Berland, M. and Charniak, E.: Finding parts in very large corpora. In Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics. New Brunswick NJ (1999)
2. Girju, R., Badulescu, A., and Moldovan, D.: Learning semantic constraints for the automatic discovery of part-whole relations. In Proceedings of the Human Language Technology Conference, Edmonton, Canada (2003)
3. Hearst, M. A.: Automatic acquisition of hyponyms from large text corpora. In Proceedings of the Fourteenth International Conference on Computational Linguistics, Nantes, France (1992)
4. Kifer, M., Lausen, G., Wu, J.: Logical foundations of object-oriented and frame-based languages. In Journal of the ACM (JACM), 42(4):741-843 (1995)
5. Kruengkrai, C., Charoenporn, T., Sornlertlamvanich, V., Isahara,H.: Acquiring selectional preferences in a thai lexical database. In Proceedings of the 1st Joint Conference on Natural Language Processing (IJCNLP-04), China (2004)
6. Li, H., and Abe, N.: Generalizing case frames using a thesaurus and the MDL principle. Computational Linguistics, 24(2): 217—244 (1998)
7. McCarthy, D.: 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 (2000)
8. Rissanen, J., and Ristad, E.: Language acquisition in the mdl framework. In Eric Sven Ristad, Language Computation. American Mathematical Society, Philadelphia (1994)
9. Sundblad, H.: Acquisition of hyponyms and meronyms from question corpora. In Proceedings from the Workshop on Natural Language Processing and Machine Learning for Ontology Engineering, Lyon, France (2002)
10. Wagner, A.: Enriching a lexical semantic net with selectional preferences by means of statistical corpus analysis. In Proceedings of the WS on Ontology Learning, (2000)