MATERIALS AND METHODS

Materials

1. Computer

The algorithms are implemented by using Visual Basic and PHP programming language. The experiments are run under the following computer specification.

- Pentium processor 1.73 GHz.
- RAM 1 GB
- Hard Disk 50 GB

2. Data

This research proposes the methodologies for automatic ontology building with a variety of resources i.e. text, thesaurus and dictionary.

2.1 Text

The corpus used to test the methodologies in this work deals with the domain of agriculture. It is the plain text in Thai and its size is of 302,640 words from 90 documents. The documents coming mainly from two resources as follows:

- Technical documents about plants from the Department of Agriculture and the Department of Agricultural Extension: It contains about 277,164 words from 85 documents.

- Thai encyclopedia on topic of plants: Its size is of 25,476 words from 5 documents.

2.2 Thesaurus

In this research, we emphasize to utilize AGROVOC Thesaurus (FAO, 2007), which is a multilingual agricultural thesaurus in English, French, Spanish, Portuguese, etc. that developed and maintained by the Food and Agriculture Organization (FAO) of the United Nations. It contains 16,607 descriptors and more than 10,706 non-descriptors (synonyms). It is used for indexing and searching information resources within the agricultural domains such as plant, fisheries, food, etc.

2.3 Dictionary

The system was based on the “Thai Plant Names” Dictionary, which developed by Smitinand and edited by the Forest Herbarium Royal Forest Department in 2001. It contains 37,110 words.

Methods

Figure 22 shows the overview of the Automatic Thai Ontology construction and Maintenance System consisting of Ontology Extraction, Ontology Integration and Reorganization and Verification. There are three sources for ontology extraction: unstructured (raw) texts, a semi-structured dictionary and a structured thesaurus. Unstructured texts should be dealt with by a hybrid approach: natural language processing, rule based and statistical based techniques being used in concert for identifying the related ontological terms and their relationships. For the semi-structured dictionary, only a task-oriented parser is needed to extract the terms and relations. However, the parser will work well if, and only if, the dictionary has a given structure. Since ontological terms can be transformed straightforwardly in the case of a structured thesaurus, we must make sure to have clean relationships between terms and possibly make certain refinements. However, even if we have gotten the appropriate ontological relationships, we still need to perform natural language processing at the phrase level and rely on machine learning techniques. Finally, all sub-ontology trees are integrated to the core-ontology by using term matching technique. The ontology will be reorganized by pruning the redundancy relationships and merging the similar concepts. In addition, we develop tool for expert to verify and extend the Ontology.

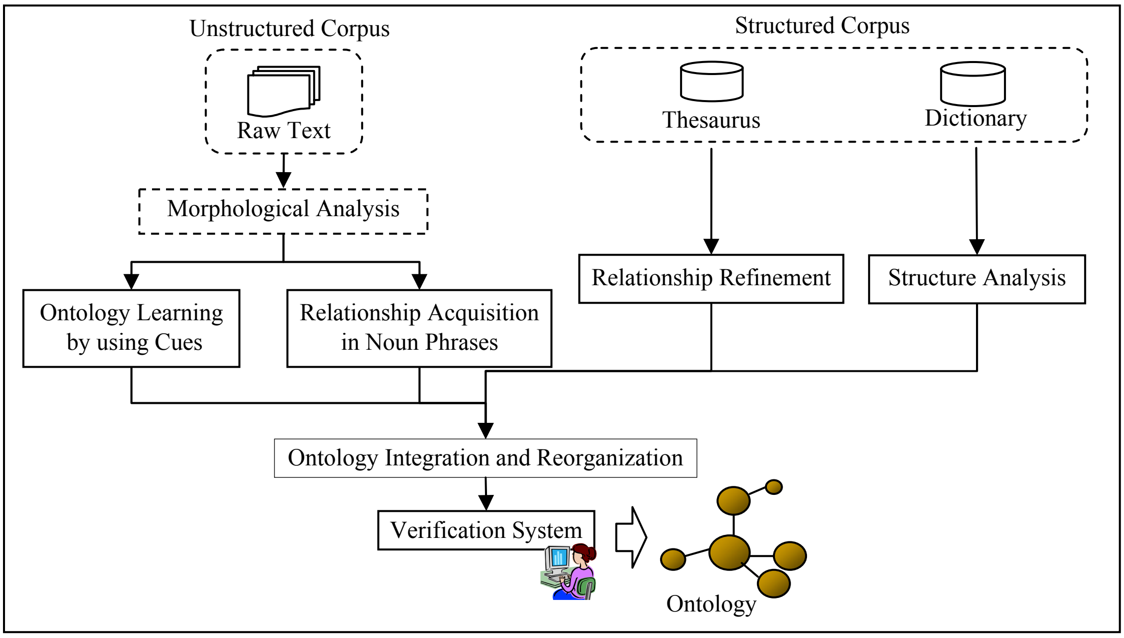


Figure 22 A conceptual framework of ontology construction and maintenance system

1. Ontology Learning based on Unstructured-Text

*Morphological Analysis and NPs Chunking***.** Similarly to many other Asian languages, in Thai there are no delimiters (blank space) to show word boundaries. Texts are a single stream of characters. Hence, word segmentation and part-of-speech (POS) tagging (Sudprasert, 2003) are necessary for identifying a term unit and its syntactic category. Once this is done, sentences are chunked into phrases (Pengphon, 2002) to identify noun phrase boundaries. In this paper, the parser relies on Noun Phrase (NP) rules, word formation rules, and lexical data. The accuracy of compound noun grouping is 92% and the accuracy of NP analysis with word formation is 90%. Before experimenting in the next process, the experts verified and corrected all the

NPs in the documents in order to test the actual performance of the ontological learning system.

1.1 Ontology Learning by using Cue (Imsombut and Kawtrakul, 2007)

Figure 23 gives an overview of the ontology learning by using cues which are lexico-syntactic pattern and item list. As far as the ontology learning is concerned, there are three main processes involved: ontological-element (concept and relation) identification by using cues: lexico-syntactic patterns and item list, candidate term generation, and candidate term selection.

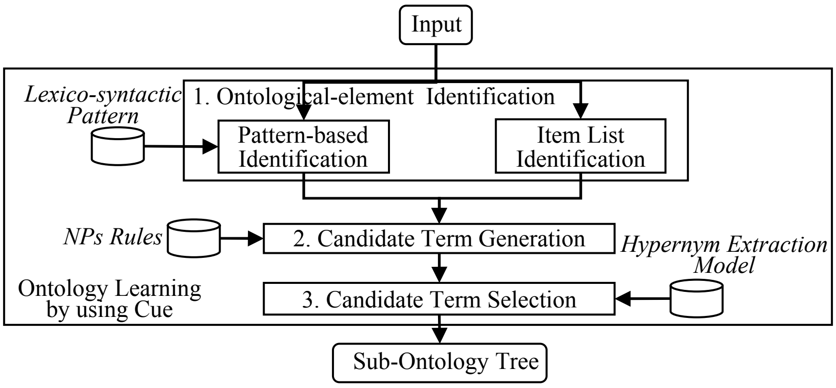

Input
Lexico-syntactic Pattern
1. Ontological-element Identification
Pattern-based Identification
Item List Identification
NPs Rules
2. Candidate Term Generation
Hypernym Extraction Model
Ontology Learning by using Cue
3. Candidate Term Selection
Sub-Ontology Tree

Figure 23 Architecture for building and maintaining an ontology of Thai

1.1.1 Ontological-Element Identification

We identify the ontological-element (concepts and relations) hinted by cues, which are lexico-syntactic patterns and item lists.

a. Pattern-based Identification

In order to collect hypernym relation patterns, we use IS-A relations with words (word pairs) from the AGROVOC thesaurus (FAO, 2007). From agricultural documents, we extract all sentences that contain the selected word pairs.

Finally, we manually select lexico-syntactic patterns from sentences by considering only the ones that the structure occurs often.

Table 12 Lexico-syntactic patterns with frequency of occurrence

Patterns	Cue-word meaning	POS	Occurrence Frequency	%
NP_0 …/chen/… NP_1, …,NP_n	(for example)	conjncl	392	41
NP_1… /pen/ NP_0	(is/am/are)	vcs	306	32
NP_0 .../daikae/… NP_1, …, NP_n	(i.e.)	conjncl	186	19
NP_0 .../champhuak/… NP_1	(kind of)	ncn	40	4
NP_1, …,NP_n /lea/ NP_0 /uen uen/	(and other)	conj	22	2
Other Patterns	-	-	16	2

Remark: vcs = complementary state verb, conjncl = noun clause conjunction,
conj = conjunction, ncn = common noun.

The most occurring ones are focused (the top-5 patterns) in this article. By using the patterns above (see Table 12), the sentence anchoring process could identify plausible sentences whose content bare its ontological relation.

b. Ontological-list Identification

In this process, we propose a methodology for identifying an ontology element from item lists that we focus on bullet list and numbering list. Since item lists could be used to describe objects, procedures, and the like, this might lead to non-taxonomic lists. In order to identify object lists which contains ontological terms, the items of the list should be the Named Entity (NE), recognized by NER system (Chanlekha and Kawtrakul, 2004). Applying NER for identifying ontological lists works well in technical domain such as agriculture and bio-informatics since the growth of ontological terms almost come from the new entities. As shown in Figure 16 and 17, it still has two problems: long boundary description and embedded list that cause the ambiguity of item list members. In order to solve these problems, the items that use the same bullet symbol and have same NE class will be considered as the same list. Like the bullet list, the items in the ordering number and having the same NE class will be considered as the same list.

1.1.2 Candidate Term Generation

In this work, we use linguistic information in the form of a grammar that mainly allows NPs to be extracted as candidate terms. Thus, this process checks whether some NPs occurred before the cue as candidate terms in order to generate the corresponding ontological terms. Thus, all NPs on the left hand side of the cue word in the pattern are generated as the candidate terms and the terms that occur in the preceding paragraphs of the item list are candidate hypernym terms. To do so it will consider only NPs corresponding to the NP's grammatical rules as shown in Table 13.

Table 13 Grammatical rules of noun phrases for ontological terms extraction.

Pattern	**Example**
NP2 = (ncn|nct+ncn|npn) + NP$^?$	[/chuea/(pathogen):ncn /wairat/(virus):ncn] (virus pathogen)
NP3 = NP2 + adj	[/kulap/(rose):ncn /daeng/(red):adj] (red rose)

NP4 = NP + VP
VP = vi|(vt+NP)
	[/a-ngun/(grape):ncn /tham/(produce):vi /wai/(vine):ncn] (vine grape)

NP5 = NP + PP
PP = prep + NP
	[/sinkha/(product):ncn /caak/(from):prep /tangprathet/ (foreign country):ncn] (product from foreign country)

Remark:			
Remark:	adj = Adjective	ncn = Common noun	nct = Collective noun
	npn = Proper noun	NP = NP2|NP3|NP4|NP5	prep = Preposition
	PP = Prepositional Phrase	vi = Intransitive verb	vt = Transitive verb
	VP = Verb Phrase	x|y = either x or y	x + y = x precede y
	x$^?$ = x can occur 0 or 1 time		

Even there are many NP rules, some NPs could not be an ontological term such as [ncn+conj+ncn], [ncn+det], where conj is conjunction and det is determiner. For example, [*/phak/(vegetable):ncn /lae/(and):conj /phonlamai/(fruit) :ncn*] (*vegetable and fruit*). The selected ontological terms should be separated into two terms, i.e. */phak/(vegetable)* and */phonlamai/(fruit)*.

1.1.3 Ontological Term Selection

Having generated the ontological candidate terms, the system will select the ontological term from a set of candidates. The most likely hypernym value (*MLH*) of term in the candidates list will be computed on the basis of an estimated function taking lexical and co-occurrence features into account. Let $h_i \in H$, H is the set of candidates of possible hypernym terms, while t_j is the related term j which is the term on the right hand side of lexico-syntactic pattern or the term in the item list. The features of the learning system are lexical and co-occurrence features. The estimate function for computing the most likely hypernymy term is defined as follows:

$$MLH(h_i,t_j) = \alpha_1 \cdot f_1(h_i,t_j) + \alpha_2 \cdot f_2(h_i,t_j) + \ldots + \alpha_n \cdot f_n(h_i,t_j) \qquad (11)$$

Where α_k is the weight of feature k, f_k is the feature k, t_j is the related term j and n is total number of features (here, we use 5 features). f_1-f_4 are lexical features and f_5 is co-occurrence feature. The system will select the candidate term that has the positive and maximum *MLH* value in each candidate set to be the ontological term of the related terms.

For weighting each feature, we test with several techniques: Information Gain, Information Gain Ratio and SVM (with linear kernel) and we found that the feature weights from Information Gain Ratio gives the best result in the candidate term selection process. Information Gain Ratio is introduced to compensate the bias of the Information Gain. They are used to decide which features are the most relevant.

However, calculating information gain needs discrete value but the co-occurrence feature (f_5), is continuous value then it needs the method to convert continuous value to discrete value that is described in the detail of feature 5.

Features and their details are following.

f_1: Head word compatibility. This feature evaluates whether head word of candidate term is compatible with head word of related term or not.

$$f_1(h,t) = \begin{cases} 1 \text{ if } h \text{ is compatible with the head word term of } t. \\ 0 \text{ if otherwise.} \end{cases} \qquad (12)$$

If the head word of a constituent is identical to the head word of another constituent, then these terms are related to each other. For more details, consider the following example.

(15) /**po-kra-chao** thi niyom pluk kan nai **prathed-thai** mi 2 chanit dai-kae **po-krachao-fak-yao** lae **po-krachao-fak-krom**/

(*There are 2 kinds of* ***Jute*** *generally planted in* ***Thailand*** *i.e.* ***Tossa Jute*** *and* ***White Jute***).

In this example, the candidate terms are *Jute* and *Thailand*. The head word of *Tossa Jute* and *White Jute* is *Jute,* then *Jute* has more possibilities to be an ontological term than *Thailand*.

f_2: NE class. This feature evaluates whether a candidate term of a hypernym belongs to the same NE class as related term or not.

$$f_2(h,t) = \begin{cases} 1 \text{ if } h \text{ belongs to the same NE class as } t. \\ -1 \text{ if } h \text{ belongs to the different NE class as } t. \\ 0 \text{ if otherwise.} \end{cases} \tag{13}$$

We consider the NE class as the feature because the cue word /pen/ might occasionally have the meaning “has symptom as”. For example,

*(16) */kalampli <u>pen</u> rok-naole/*
(Cabbage <u>has symptom as</u> Soft-Rot.).

Here, *Cabbage* and *Soft-Rot* are NEs which have different classes, i.e. plant and disease, respectively. Accordingly, *Soft-Rot* is not a hypernym of *Cabbage*. In addition, we classify this feature’s value to three values, i.e., 1,-1 and 0, where 0 is assigned for the terms being at a high level of the taxonomy, e.g. /phuet trakun thua/(pulse crops) which are not NE.

f₃: Property list term. Since the cue word /pen/ (be) can be used to express the properties of an object. For example,

*(17) */kap-bai <u>pen</u> si-namtan/*
(The *leaf <u>is</u> brown-color.*)

Brown-color is not the hypernym of *leaf*, but a property of the object leaf. This being so, we defined a set of properties to be able to determine which terms are concepts and which are properties. In the domain of agriculture, there are 3 types of property lists: colors, shapes, and appearances; e.g. *powder*.

$$f_3(h,t) = \begin{cases} -1 \text{ if } h \text{ is a property term.} \\ 0 \text{ if otherwise.} \end{cases} \tag{14}$$

f₄: Topic term. This feature evaluates whether candidate term is the topic term of the document (short document) or a topic term of the paragraph (long document) or not. Here, topic term will be computed by using tf*idf where tf is the term frequency and idf is inverse document frequency (Salton, 1989).

$$f_4(h,t)=\begin{cases}1 \text{ if } h \text{ is a topic term of the document (short document)}\\ \quad\text{or a topic term of the paragraph (long document).}\\ 0 \text{ if otherwise.}\end{cases} \tag{15}$$

f_5: Co-occurrence feature. Some statistical methods are used to analyze the co-occurrence of the candidate and the related terms. We explore three alternatives, Mutual Information (MI) (Church and Hanks, 1989), log-likelihood ratio (LL) (Dunning, 1994), and Chi-square testing (χ^2). By experimenting with Thai agriculture document, we found that Chi-square has the highest precision. Chi-square is based on hypothesis testing. It measures the divergence of the observed and expected data. Chi-square can be defined as follows:

$$f_5(h,t)=\chi^2=\frac{N(ad-bc)^2}{(a+b)(a+c)(b+d)(c+d)} \tag{16}$$

Where *a* represents the frequency of the term *h* occurring in the same sentence with the term *t*. The value *b* (respectively *c*) is the number of occurrences of term *h* (respectively t) in the corpus for sentences not containing term t (respectively h). The value *d* indicates the number of sentences that do not contain neither *h* nor *t*. The total number of sentences in the corpus is represented by *N*.

Since value for calculating Information Gain must be discrete value and the result of chi-square is continuous value, then the method for partitioning the continuous value to a discrete value is needed. We partition this feature value into two intervals at value *x*. However, chi-square value is very sparse. From observation, the minimal value of chi-square of our corpus is 0.001 and the maximum value is 206.667. Partitioning the data into two intervals by using this minimal and maximum value is not appropriate because in some candidate sets the chi-square values of all data are very low and this cut point can not separate data between positive and negative class. Then we define *x* depending on each candidate term set as *x* = (*Max*+*Min*)/2 where *Max* and *Min* are the maximum and minimum chi-square value in the candidate term set. This partition value will separate data into two groups (≤ *x* and >*x*; f_5=0 and f_5=1) for calculating Information Gain. Figure 24 shows an example sentence, the calculation of the partition value of this sentence and the co-occurrence feature of each candidate term. Table 14 shows examples of sentence, the candidate terms and their feature vectors and *MLH* values for selecting the ontological term.

Last ***year****, a lot of* ***roses*** *have been imported from* ***abroad*** *such as variety of* ***Sacha****,...*

$$\left.\begin{array}{l} \chi 2(year,\ Sacha) = 0.263 \\ \chi^{2}(roses,\ Sacha) = 61.999 \\ \chi 2(aboard,\ Sacha) = 36.315 \end{array}\right\} x = \frac{(0.263+61.999) = 31.129}{2} \Rightarrow \begin{array}{l} f_5(year,\ Sacha) = 0 \\ f_5(roses,\ Sacha) = 1 \\ f_5(aboard,\ Sacha) = 1 \end{array}$$

Figure 24 An example of calculation for the co-occurrence feature

Table 14 Examples of sentence, the candidate terms and their feature vectors and MLH values

	Head word	NE	Property Term	Topic Term	Co-occur.	MLH Value
weight	0.001	0.069	0.139	0.014	0.029	
Last ***year****, a lot of* ***roses*** *have been imported from* ***abroad*** *such as variety of* ***Sacha, Mercedes*** *and* ***Gabrielle****.*						
Year,Sacha	0	0	0	0	0	0
Rose,Sacha	0	1	0	1	1	**0.112**
Aboard,Sacha	0	0	0	0	1	0.029
There are 2 kinds of ***Jute*** *generally planted in* ***Thailand*** *i.e.* ***Tossa Jute*** *and* ***White Jute****.*						
Jute, Tossa Jute	1	1	0	1	1	**0.113**
Thailand, Tossa Jute	0	0	0	0	0	0
Cabbage *has symptom as* ***Soft-Rot****.*						
~~*Cabbage, Soft-Rot*~~	0	-1	0	1	0	-0.55

The weighting of each feature in Figure 14 will be discussed in Results and Discussion chapter. The ontological term of each sentence is the candidate term that has the positive and maximum MLH value. The sentence that has the negative *MLH* value of candidate term will be pruning.

1.2 Relationship Acquisition in NPs (Imsombut and Kawtrakul, 2005)

The information concerning semantic relations can be extracted not only at the sentence level, but also at the noun phrase level. In Table 15, we list the semantic relations of NPs our system is able to analyze by taking as input a Thai corpus in the domain of agriculture. The semantic relations in the list are the most frequently ones found in the data, and they are based on relations given in (Vanderwende, 1994; Barker and Szpakowicz, 1998; Soergel, 2004; Kawtrakul *et al.*, 2005). Even though our analysis is on texts dealing with agriculture, the semantic relations are domain independent. An overview of learning system for discovery semantic relations in NPs is shown in Figure 25.

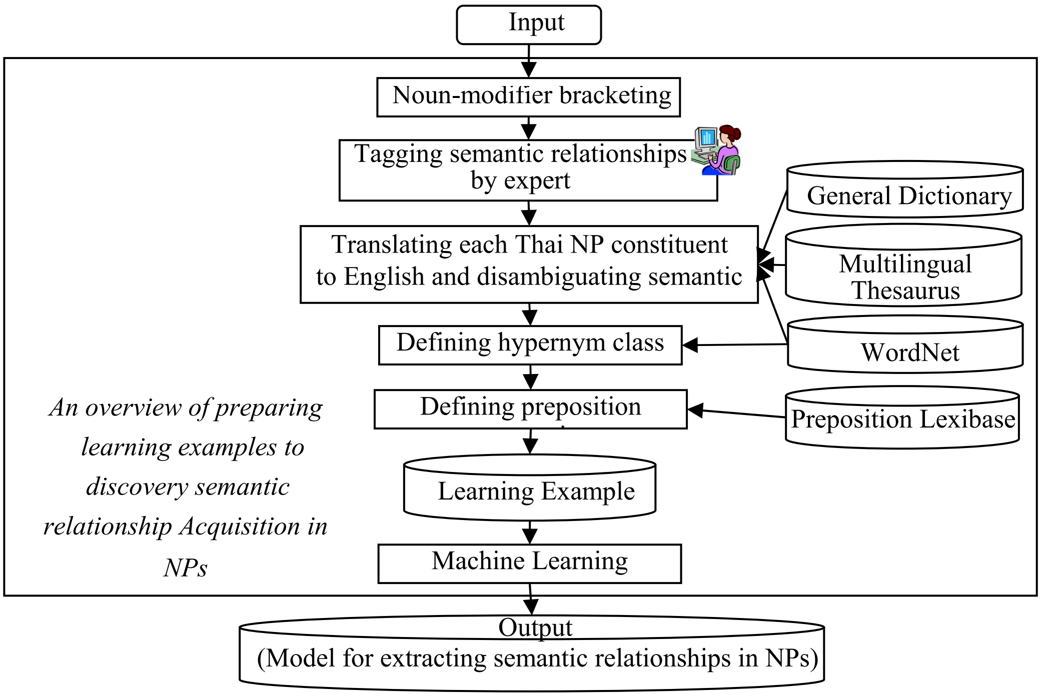

Input
Noun-modifier bracketing
Tagging semantic relationships by expert
General Dictionary
Translating each Thai NP constituent to English and disambiguating semantic
Multilingual Thesaurus
Defining hypernym class
WordNet
Defining preposition
Preposition Lexibase
An overview of preparing learning examples to discovery semantic relationship Acquisition in NPs
Learning Example
Machine Learning
Output
(Model for extracting semantic relationships in NPs)

Figure 25 An overview of learning system for discovery semantic relations in NPs

Table 15 A list of semantic relations.

Relationships	Examples
1. **IS-A**	/ka-fae/(coffea) /ro-bus-ta/(Robusta) : (Robusta coffea)
2. Location	/phuet/(plant) /khet-ron/(tropical area) : (tropical plant)
3. **Purpose** :	/cream/(cream) /sa-lad/(salad) : (salad cream), /khai-mun/(fat) /sam-rub/(for) /prung/ (cook) /a-han/ (food) : (cooking fat)
4. **Possessor**	/rai/(field) /ka-set-ta-kon/(farmer) : (farmer field), /ran-kha/(shop) /khong/(belong_to) /chum-chon/(community) :(community shop)
5. **Made-of**	/sous/(sauce) /tua-luang/ (soybean) : (soybean sauce), /nuy/(cheese) /jak/(from) /nom/(milk) /kea/(sheep) : (sheep cheese)
6.**Source (from)**	/mun/(dung) /sat/ (animal) : (dung), /dok-mai/(flower) /jak/(from) /tang-pra-ted/ (foreign_country) : (flower from foreign country)
7. **Topic**	/kor-mul/(data) /pra-mong/(fishery) : (fishery data), /kor-mul/(data) /tang/(-) /pum-mi-ar-kart/(weather) : (weather data)
8. **Property**	/ku-lab/(rose) /see-deang/(red_color): (red rose)
9. **Part-Whole**	/perk/(husk) /kao/(rice) : (rice husk), /kao/(horn) /kong/ (belong_to) /sat/(animal) : (horn)
10. **Container**	/klong/(carton) /nom/(milk) : (milk carton)

During this step, the system will extract semantic relations from Noun Phrases (NPs) by learning the common ancestral concept of their head and modifier. The following features are taken into account by our learning component:

1. the semantic class of the head noun: The system will extract the head noun's sense and its hypernyms by using WordNet.

2. the semantic class of the modifier noun or head noun of a modifier phrase: Like the head noun, the system will extract the sense of the modifier noun or head noun of a modifier phrase and its hypernyms with the help of WordNet.

3. the semantic class of the preposition: This is applied only to NPs composed of a prepositional phrase. It is meant to provide information about the semantic role of the prepositions used in the NPs. The value of this feature is determined by Lexibase (Kawtrakul, 2004), a resource developed in our laboratory.

Since our learning features are based on the semantic information provided by WordNet, the learning examples have to be translated from Thai to English. To this end our system brackets first the head and the modifier of a given NP, to translate it into English then using a Thai-English Thesaurus, AGROVOC (FAO, 2007), and a Thai-English Dictionary, LEXiTRON (NECTEC, 2007). Next, the WordNet sense of nouns and the semantics of prepositions are identified. Here are some details concerning the algorithm.

1.2.1 Head noun and modifier segmentation

This step is similar to the approach taken by (Lauer, 1995; Barker, 1998). For a given sequence of X-Y-Z, segmentation is determined by comparing occurrences of X-Y with occurrences of X-Z in a corpus. If X-Z occurs in the corpus then the segmented phrase is [X-Y]-Z, otherwise it is X-[Y-Z]. If a phrase like '*/kuad/(bottle) /nam/(water) /plad-sa-tik/(plastic): (plastic water bottle)*', '*/nam/ /plad-sa-tik/ (water plastic)*' never occurs in a corpus then it will be segmented as [[*/kuad/(bottle) /nam/(water)*] */plad-sa-tik/(plastic)*].

If the phrase '*/kuad/(bottle) /nam/(water) /phon-la-may/(fruit): (fruit juice bottle)*', '*/nam/ /phon-la-may/ (fruit juice)*' occurs in a corpus then it will be segmented as follows [*/kuad/(bottle)* [*/nam/(water) /phon-la-may/(fruit)*]].

1.2.2 Translating each Thai NP constituent into English and disambiguation of the semantics.

At this step, the system will translate all constituents of Thai NPs. If the constituent was a group of words, then system will, translate it first into an English word. For example, [*/nam_phon-la-may/(water_fruit)*] is translated into '*fruit_juice*'.

If there is no such word in English, the system will translate only the head of the word group. For example, [*/kuad_nam/(bottle_water)*] will yield '*bottle*'. There are two techniques to accomplish this task.

a. Technique 1

The system uses a Thai-English thesaurus to translate a Thai word (*tw*) into its English correspondence (*ew*), every words having a one-to-one mapping, i.e. translation equivalent. However *ew* might have more than one word sense in WordNet. In such a case the system needs to disambiguate the senses by computing the most likely similarity between each sense *i* and the hyponyms of *ew* in a thesaurus.

$$ew^{i} = \underset{ew^{i}}{\arg\max} \sum_{j=1}^{n} similarity\ (ew^{i}, h_{j}) \tag{17}$$

$h_j \in Hyponym(ew)$

similarity(x,y) = the amount of edges in common paths between x to root and y to root

Figure 26 shows an example of thesaurus-based semantic disambiguation of '*/phon-la-may/ (fruit)*'. Fruit has three senses in WordNet. Based on using thesaurus information, it can be decided that sense number one is the sense of fruit in the domain of agriculture. If the word does not exist in the thesaurus, it will be processed by using technique 2.

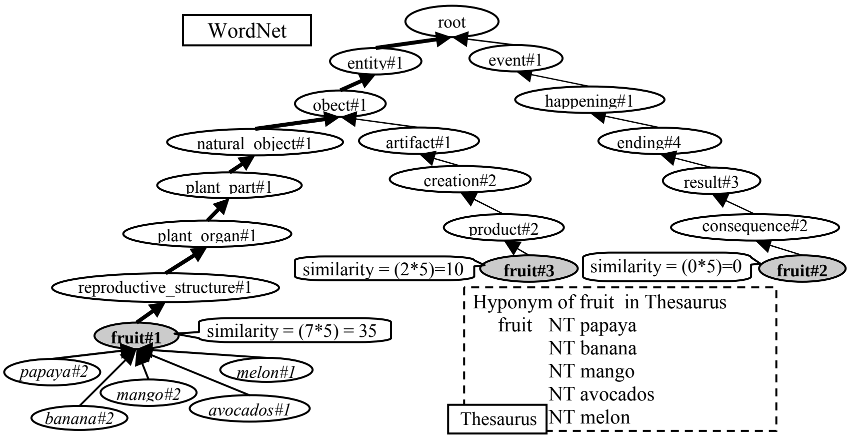


Figure 26 An example of thesaurus-based semantic disambiguation of ‘*/phon-la-may/(fruit)*’.

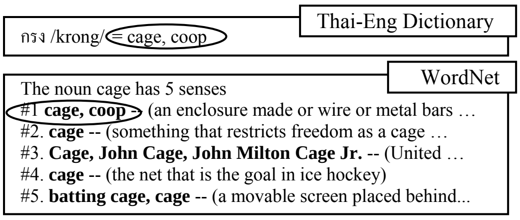
Remark. word#x stand for word in sense x.

b. Technique 2

A Thai-English dictionary is used to translate a Thai word (*tw*) into English (*ew*). By doing, we found that there are a lot of Thai words with a given meaning that could be translated into several words in English (about 70%). In this case we take the first English word or one of its synonyms in the dictionary as we assume that this is the most frequently one used. This word is then compared to the words in WordNet. If there are several senses in WordNet, the system will select the sense with the highest number of ‘synset’ terms similar to the set of translated words. Let S^{ew^i} be a synset i^{th} of *ew* in WordNet and T be the set of translated terms of *tw* in a dictionary. In this case the system will select the *ew* sense by using the following equation.

$$ew^j = \underset{ew^j}{\arg\max}\, sizeof(S^{ew^j} \cap T) \qquad (18)$$

Figure 27 shows an example of a dictionary-based semantic disambiguation of ‘/*krong*/ (cage)’



The noun cage has 5 senses
#1 **cage, coop** -- (an enclosure made or wire or metal bars …
#2. **cage** -- (something that restricts freedom as a cage …
#3. **Cage, John Cage, John Milton Cage Jr.** -- (United …
#4. **cage** -- (the net that is the goal in ice hockey)
#5. **batting cage, cage** -- (a movable screen placed behind...

Figure 27 An example of dictionary-based semantic disambiguation of '/krong/(cage)'

If the Thai word had several meanings, or if it had only one translation, then the system will alert the expert to select the sense of the word in WordNet, manually.

In the case where the NP is composed of various words absent from WordNet, the system will select only the head noun for semantic disambiguation. For example, for the term '*/tang-pra-ted/(foreign country)*' the system will choose only '*country*' in '*foreign_country*' to disambiguate it semantically.

1.2.3 Defining hypernym class and preposition semantic

After translating and disambiguating the semantics of the head and the modifier of the NP, we extract from WordNet the complete hypernym path of each constituent in NP for the learning system. For example, the hypernym path of NP *"/kuad/(bottle) /nam phon-la-may /(fruit-juice): (fruit-juice bottle)*"

{bottle#1, vessel#3, container#1, instrumentality#3, artifact#1, object#1, entity#1}

{fruit_juice#1,beverage#1, food#1, substance#1, entity#1}

Moreover, for NPs containing a preposition, for example "*/dok-mai/(flower)* ***/jak/(from)*** */tang-pra-ted/ (foreign_country):(flower from foreign country)*", the system will define the semantic group to which it belongs by using Lexibase (Kawtrakul, 2004). In Thai, there are 10 semantic groups of prepositions such as location, purpose, time, etc. These semantic groups are mapped into 10 features. The feature values are set to 1 if the preposition has the same semantic group as the feature. Some prepositions belong to several semantic groups such as '*/nai/ (in)*" which can express a location or time meaning. Hence, the system will set the value of these features to 1. For NPs without prepositions, all values of these features are 0.

1.2.4 Learning of relationships

To obtain vectors of equal length, all hypernym list class of all examples are union to be list of hypernym class and the features will be converted into binary representations. Then, the features vector are the list of hypernym class of head, list of hypernym class of modifier and the list of the semantics, i.e.

features_vector{{list of hypernym class of head}, {list of hypernym class of modifier}, {list of preposition semantic}}

The features will be converted into binary representations to obtain vectors of equal length. The learning system will be applied to learn the common ancestral concept of the head and the modifier, to generate then a model to extract the semantic relationship of the NPs. Two machine learning techniques are applied by our system.

1) C4.5 of decision tree learning system by using the software package of Weka (The University of Waikato, 2007).

2) Support vector machine: We use several kernels but linear kernel is shown the best result. The software we used is the LIBSVM package (Laird, 2005).

For the experimental results, the decision tree learning system generated around 90 classification rules for discovery 10 semantic relations as mentioned in Table 3. For examples:

Rule: If head in container#1 then rel. Container.

Ex. */kuad/(bottle) /nam phon-la-may /(fruit-juice): (fruit-juice bottle)*

By applying this rule, the semantic relationship of noun "*/krong/(carton)*" and "*/nom/(milk)*" in "*/krong/(carton) /nom/(milk):(milk carton)*" will be '*container*' relation since carton is in the class of container#1.

In addition, the SVM learning system generated 10 learning models accordingly to the relations. The experimental results are shown in the next chapter.

2. Ontology learning based on a thesaurus

As mentioned in Section 2, the data of AGROVOC (FAO, 2007) should be cleaned before being used for ontology construction. We have divided this process of data cleaning and refinement of semantic relations into three main steps: Acquisition of Refinement Rules, Detection and Suggestion, and, finally, Verification. A system overview is given in Figure 28. (Kawtrakul *et al.*, 2005)

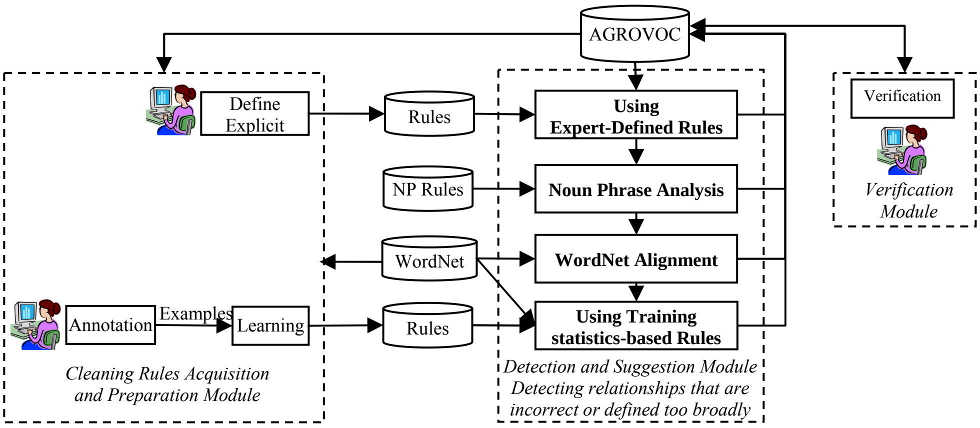

AGROVOC
Define Explicit
Rules
Using Expert-Defined Rules
NP Rules
Noun Phrase Analysis
WordNet
WordNet Alignment
Annotation
Examples
Learning
Rules
Using Training statistics-based Rules
Cleaning Rules Acquisition and Preparation Module
Detection and Suggestion Module Detecting relationships that are incorrect or defined too broadly
Verification
Verification Module

Figure 28 The process of cleaning and refining term relationships

2.1 The Rule Acquisition Module: Expert-defined Rules and Learning by Example

To mine the implicit relationships of some NPs, this module acquire a set of rules to suggest the most likely relationships in case that the relationship given by AGROVOC is underspecified (defined too broadly), especially RT. The rules will be provided by experts and by machine learning.

2.1.1 Expert-defined Rules

The experts can simply define a set of rules for allowing the correction of inappropriate relationships. They observe AGROVOC's data and define rules using data concerning concept types given in AGROVOC's database. For example, the rules constraint consists of the data in 'concept type data', the category of terms such as GC (Geographic term: Country level), and TP (Taxonomic term: Plant).

Given these rules, a relationship satisfying them will be revised automatically. For example, consider the following rule:

If X and Y are marked as ***"TP"*** *in the concept type field, and if X* ***BT*** *Y, then X<subclassOf> Y*

According to AGROVOC, the concept types of *Rosaceae* and *Malus* are TP, related by **BT**. Hence, the original relationship BT of "*Malus* BT *Rosaceae*" will be replaced by the <*subclassOf*>.

2.1.2 Learning rules-from-Examples

In this case, the rules are prepared to learn from examples in order to refine a relationship called ***RT.***

To prepare the learning set, we provide an annotation tool allowing the domain expert to manually tag term senses (labeled by a sense id number in WordNet). It allows also to specify the appropriate semantic relationship between some terms, for example, (*Sheep#1 <usedToMake> Mutton#1*).

In the case of compound nouns, only the noun heads are used. For example: *Rice* and *Rice Flour* will be annotated as follows: (*Rice#1 <usedToMake> Flour#1*)

Having prepared the examples, their complete hypernym path will be extracted from WordNet.

{sheep#1, bovid#1, ruminant#1, mammal#1,vertebreate#1, animal#1, organism#1, living_thing#1, object#1,entity#1}
{mutton#1, meat#1, food#2, solid#1, substance#1, entity#1}

The hypernym list given above will be used as the basis of the features' vectors, i.e.

Features'_vector{{list of hypernym class of all term1},{ list of hypernym class of all term2}}

The features will be converted into binary representations, in order to obtain vectors of equal length. The learning system, C4.5, will be applied to learn the common ancestral concept for term1, e.g., *animal#1,* and term2, e.g*., meat#1*, to generate then the rules. Figure 29 shows the example of the data set for training the <*usedToMake*> relationship. Table 16 displays the revision rules learnt from the training set.

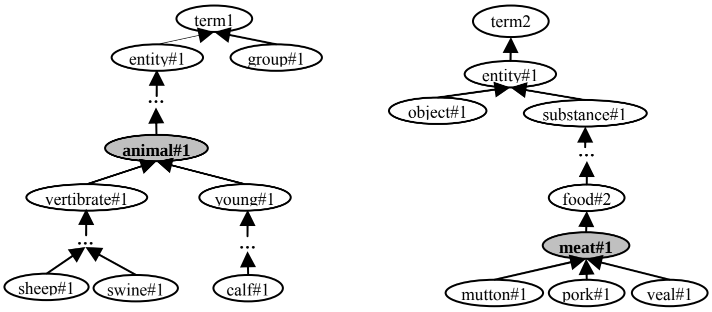


Figure 29 Examples of hierarchical data used for training the '*usedToMake'* relation

Table 16 Examples of training statistical-based rule

	Rule	**Example**
1	If class X is *animal#1* and class Y is *meat#1*, and X RT Y Then X <*usedToMake*> Y	*Sheep* RT *Mutton, Swine* RT *Pork*, *Calf* RT *Veal*
2	If class X is *plant*#2 and class Y is *food#1*, and X RT Y Then X <*usedToMake*> Y	*Rice* RT *Rice flour*, *Oat* RT *Oatmeal* , *Sugar Cane* RT *Cane Sugar*
3	If class X is *fruit#1* and class Y is *oil*#3, and X RT Y Then X <*usedToMake*> Y	*Castor beans* RT *Castor oil, Cottonseed* RT *Cottonseed oil*

By applying Rule 1, the original relationship RT of "*Chicken RT Chicken meat*" will be replaced by <*usedToMake*>.

2.2 The Detection and Suggestion Module

In this module, the system detects incorrect and inconsistently applied relationships and suggests appropriate relationships, waiting for the expert's confirmation. We propose three techniques to achieve this goal: rules for semantic relationships, noun phrase analysis, and WordNet alignment.

The outline of this algorithm is illustrated in Figure 30, where T_1, T_2 and Rel denote, respectively, Term1, Term2, and the AGROVOC relationship between them.

The relationship revision rules have been discussed in the previous section. Next, we briefly describe the procedures used for the analysis of noun phrases and the WordNet alignment**.**

2.2.1 Using Noun phrase analysis

This technique is used to analyze the surface form of a compound term's head word. If the head word of a term has the same surface form as its broader term, the system will apply the '*subclassOf*'/ '*superclassOf*' relationship. For example,

```
AGROVOC Cleaning_& Refinement ($T_1$, $T_2$, Rel)             ;Return new__relationship
Input: Term1, Term2, Relationship
Output: New Relationship
1. If (Rel = BT or Rel = NT)
   Then If Agree_Expert_defined_Rules ($T_1$, $T_2$, Rel)
        Then return new_refined_relationship.             ; following the rules
        Else If Headword-Is-Compatible ($T_1$, $T_2$)
             Then return subclass/superclass relationship.
             Else If Is_Wordnet_HypernymPath ($T_1$,$T_2$)
                  Then return subclass/superclass relationship.
                  Else If Agree_Revision_Rules ($T_1$, $T_2$, Rel)
                       Then return new_relationship            ; following the rules
                       Else return U.                          ; Un-refined
2. Else If (Rel=UF or Rel = USE)
        Then If Is_Wordnet_Synset ($T_1$, $T_2$)
             Then return synonym relationship.
             Else If Agree_Revision_Rules ($T_1$, $T_2$, Rel)
                  Then return new_relationship.                ; following the rules
                  Else return U.                               ; Un-refined
3.      Else If (Rel=RT)
             Then If Agree_Revision_Rules ($T_1$, $T_2$, Rel)
                  Then return new_relationship.                ; following the rules
                  Else return U.                               ; Un-refined
```

Figure 30 An algorithm for data cleaning and relationship refinement

Milk BT *Cow milk*

From the compound noun's analysis we see that the head word of *Cow milk* is *milk,* which obviously has the same surface form as *Milk*, the broader term of *Cow milk*. Hence, the system will apply the <*subclassOf*> relationship to *Cow milk* and *Milk*.

Milk BT *Milk fat*

The result of the analysis shows that the head word of *Milk fat* is *fat*, which is not compatible with the broader term, *Milk*. This will be detected, and the system will be trained by examples as mentioned before, in order to extract the rule for refining the relationship.

2.2.2 Using WordNet's Relationships

During this step, we use WordNet's hyper-hyponymy relationships to align the BT/NT relationship in AGROVOC. The synset of a term in WordNet is used to align the UF/USE relationship in AGROVOC. Since the relationships in WordNet are checked by experts and since it contains a great number of general, domain-specific terms, including agricultural terms, WordNet is a good resource for aligning certain relationships of AGROVOC, for example, taxonomic and synonym relationships.

At this stage the system starts retrieving the synset offset number of the AGROVOC UF/USE term in WordNet. If it can find these terms, and if they have the same synset id number, the system will consider that they are 'synonyms'. It will also check AGROVOC's broader term and the narrower term in WordNet. If it finds that the broader term is the ancestor of the narrower term in the WordNet hierarchy, it will conclude that we are dealing here with a '*subclassOf*'/ '*superclassOf*' relationship. For example,

Cabbage BT *Vegetable*

Query results for *Cabbage* and *Vegetable* in WordNet show that *Cabbage* is a hyponym of *Cruciferous vegetable,* and *Cruciferous vegetable* is a hyponym of *Vegetable*. Figure 31 shows the relationship of *Vegetable* and *Cabbage* respectively in WordNet and AGROVOC.

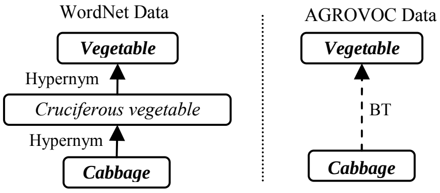


Figure 31 The relationship between '*Vegetable*' and '*Cabbage*' in WordNet and AGROVOC

Since *Vegetable* is an ancestor of *Cabbage*, the system will define *Vegetable* as a <superclassOf> *Cabbage*. In the case of *Milk* NT *Milk fat*, the relationship is not refined by this technique, because *Milk* and *fat* follow different hypernym paths in WordNet.

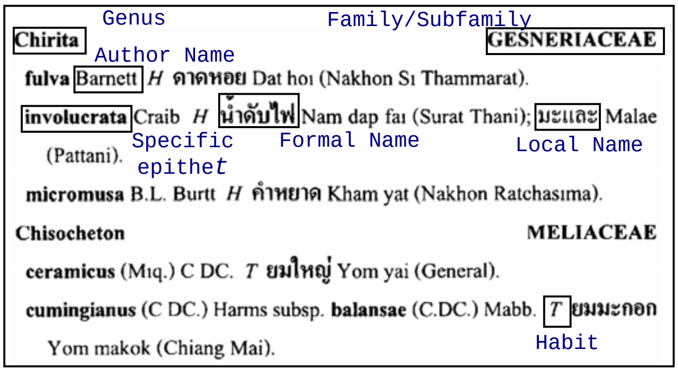
3. Ontology learning based on a dictionary

A Domain Specific Dictionary is the best way to extract relational information, as such kind of dictionary has a specific structure, as well as clear and accurate information. In this paper we use as case study the "Thai Plant Names" Dictionary, developed by Prof. Dr. Tem Smitinand and edited by the Forest Herbarium Royal Forest Department in 2001. The most frequent relationships of this specific dictionary are Hyponymy and Synonymy. Two steps are needed for extracting the ontological terms: Structure Analysis and Relation Analysis.

3.1 Structure analysis

In this system, the printed dictionary is digitalized using the optical scanner. The scanned-in image is analyzed to identify each alphabetic letter and converted into text document by OCR process (Kawtrakul and Waewsawangwong, 2000). After manually correcting OCR error, the text document is analyzed by the Task Oriented Parser to produce entity concept. The analysis of the structure of the dictionary is an important feature in order to be able to distinguish elements of word entries as sub-parts. The characteristic of term positions are analyzed and irrelevant parts, such as author name, are filtered out. The needed parts are then transferred to a relational database by using a Task Oriented Parser.

Figure 32 illustrates the analysis of the dictionary's structure. Terms are clustered by using Alphabets' characteristics (see Table 17). The position of the terms in the text, such as, top and rightmost corner, top and leftmost corner, is also considered. A relational database's fields are predefined as Hierarchical relations such as Family, Sub-Family, Genus, Specific epithet and Formal Name, respectively.


Genus
Family/Subfamily
Author Name
Specific epithet
Formal Name
Local Name
Habit

Chirita **GESNERIACEAE**

fulva Barnett *H* **ดาดหอย** Dat hoi (Nakhon Si Thammarat).

involucrata Craib *H* **น้ำดับไฟ** Nam dap fai (Surat Thani); มะและ Malae (Pattani).

micromusa B.L. Burtt *H* **คำหยาด** Kham yat (Nakhon Ratchasima).

Chisocheton **MELIACEAE**

ceramicus (Miq.) C DC. *T* **ยมใหญ่** Yom yai (General).

cumingianus (C DC.) Harms subsp. **balansae** (C.DC.) Mabb. *T* **ยมมะกอก** Yom makok (Chiang Mai).

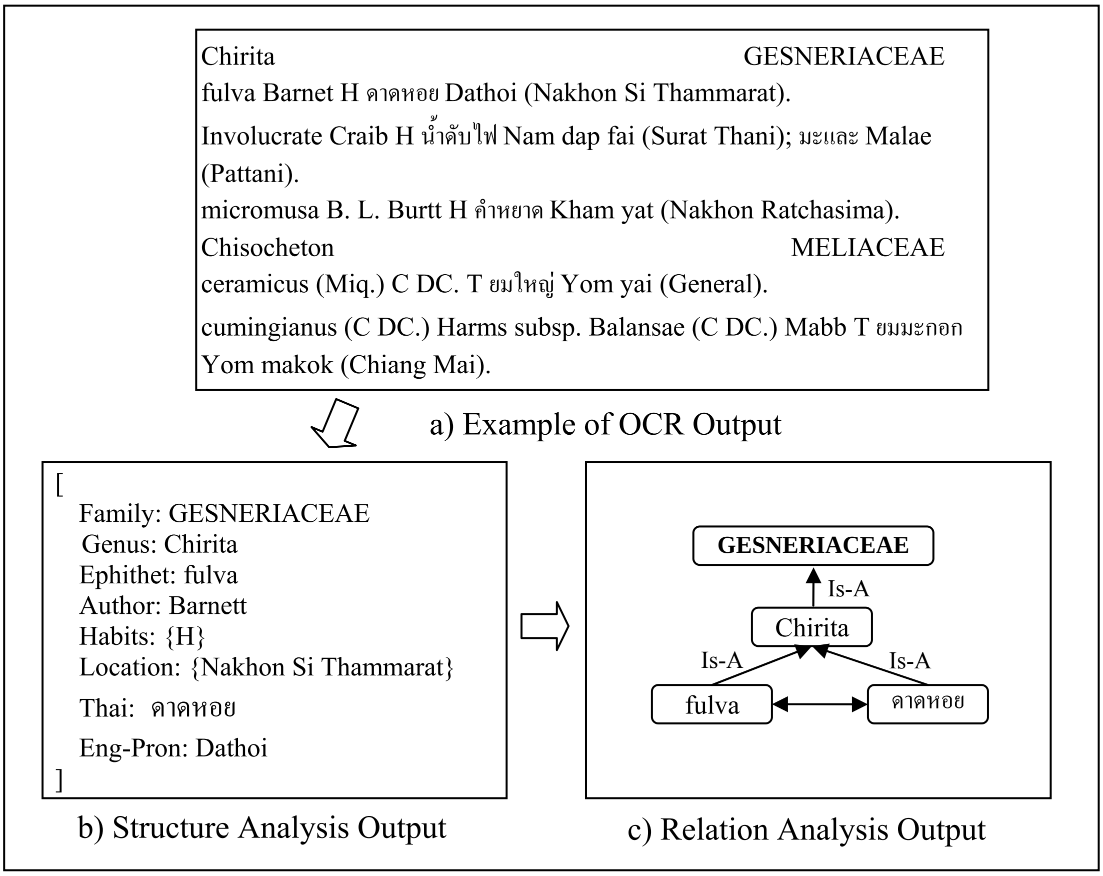
Figure 32 Dictionary structure

Table 17 Characteristics of the alphabet for dictionary conversion.

Feature	Database field	Example
All upper case at the top-rightmost corner	Family/Sub-Family	GESNERIACEAE
Starts with upper case at the top-left most corner	Genus	Chirita
All lower case	Specific epithet	involucrata
Thai alphabet in bold font	Formal Name	**น้ำดับไฟ** /Nam–dap-fai/
Thai alphabet	Local Name	มะและ /Malae/

3.2 Relation Analysis

After the parsing, Relation Analysis process will map each entity concept relation to the ontological relation. Figure 33 shows the process of ontology extraction based on specific dictionary. Figure 33a illustrates the output of the OCR system, Figure 33b shows the structure analysis output of the data in Figure 33a and the output of relation analysis is shown in Figure 33c. The system is able to extract 37,110 terms and 21,620 relationships. The experiment of dictionary-based-ontology extraction achieves 100%.



Chirita GESNERIACEAE
fulva Barnet H ดาดหอย Dathoi (Nakhon Si Thammarat).
Involucrate Craib H น้ำดับไฟ Nam dap fai (Surat Thani); มะและ Malae (Pattani).
micromusa B. L. Burtt H คำหยาด Kham yat (Nakhon Ratchasima).
Chisocheton MELIACEAE
ceramicus (Miq.) C DC. T ยมใหญ่ Yom yai (General).
cumingianus (C DC.) Harms subsp. Balansae (C DC.) Mabb T ยมมะกอก Yom makok (Chiang Mai).

a) Example of OCR Output

[
Family: GESNERIACEAE
Genus: Chirita
Ephithet: fulva
Author: Barnett
Habits: {H}
Location: {Nakhon Si Thammarat}
Thai: ดาดหอย
Eng-Pron: Dathoi
]

b) Structure Analysis Output

c) Relation Analysis Output

Figure 33 Dictionary based ontology extraction process

4. Ontology Integration and Reorganization

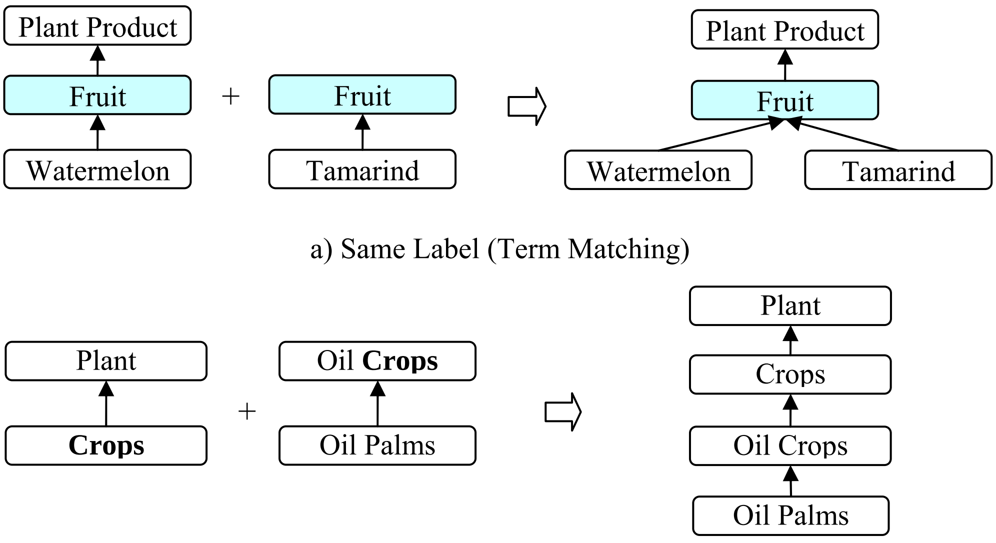
This section describes the details of the ontology integration and ontology reorganization process.

4.1 Ontology Integration

At this step, we united the related word/phrase pairs collected from our sources: Texts, some Dictionary and a Thesaurus. In order to integrate them, two heuristics are applied:

- *If the separated ontological trees have the same label nodes, then merge them.* Figure 34a shows the example of ontology integration by using this technique.

- *If the terms' head words match partially, then merge them.* The partially term's head words matching technique is shown in Figure 34b.



a) Same Label (Term Matching)

b) Partially Term's Head Words Matching

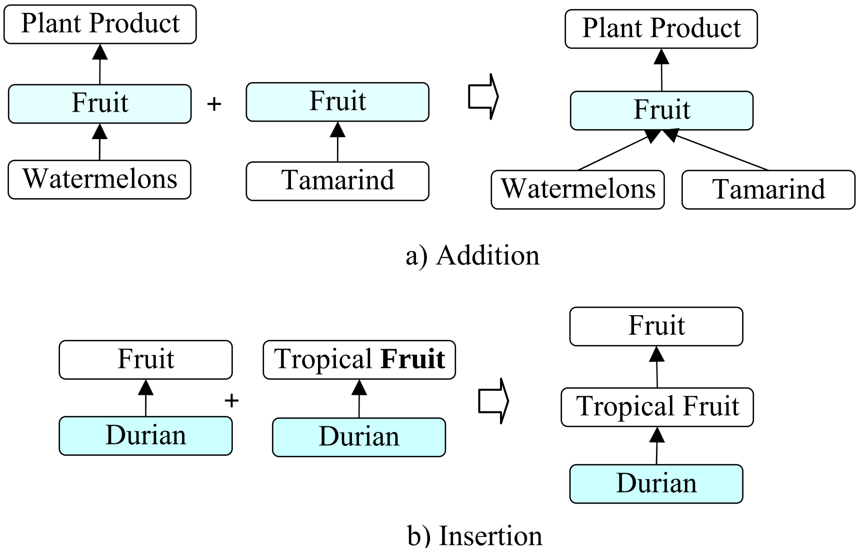
Figure 34 Techniques of ontology integration

There are two operations involved in this process of integration: Addition and Insertion. Figure 35 shows operations for ontology integration of a core-tree (left-hand-side tree) into a new ontological tree (right-hand-side tree), on the basis of information extracted from a dictionary or some raw texts.

- *Addition*: A Child node will be added to the core tree, if the parent node has the same label or partially term's head word matching to the existing node in the core tree.

- *Insertion:* If the children nodes have the same label as the head word of the parent nodes then the new, more specific term will be inserted between two existing ontological terms.

The remaining terms that could not be integrated will be kept for the expert to be added later on, manually.



a) Addition

b) Insertion

Figure 35 Operations for ontology integration

However, the ontology integration process can make the problem of conflict relationship or inconsistent relationship since the sub-ontology tree may contain an incorrect relationship. The incorrect relationship can be caused by the problem of cue word ambiguity especially the cue word /pen/ as shown in the section of problems of ontology integration. We solve this problem by comparing the occurred frequency of each relation. The assumption of this solution is that the correct relationship has more frequency than the incorrect relationship. The relation that has less frequency is deleted. This process is beneficial for pruning the incorrect relationships.

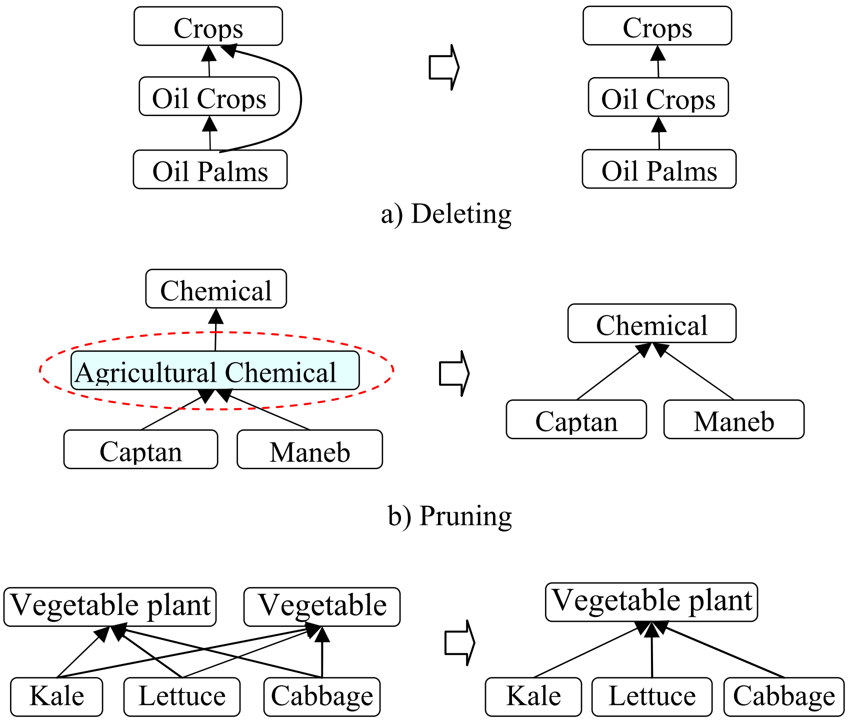
4.2 Ontology Reorganization

When all nodes and relationships in the additional ontologies are added to the core tree completely, the ontology reorganizing operation will be processed respectively. There are three operations of ontology reorganizing: deleting, pruning and merging. Figure 36 shows the example of the process of these operations.

- *Deleting:* If there are duplicate relations, the system will delete the tree with less nodes.

- *Pruning:* Node, which does not have its own property and its children is the same set as its parent, should be deleted and its children should be transferred to under its immediate parent.

- *Merging:* If the two nodes or more than two have the common set of children nodes and these node's labels are similar then these nodes are merge to the new node and the common set of children will be transferred to the new node. The similarity of node's labels are compared by using edit distance technique (Levenshtein, 1966). Furthormore, the system will select the label that are the most frequency occurred in the corpus to be the concept label or concept representation of the new node.



a) Deleting

b) Pruning

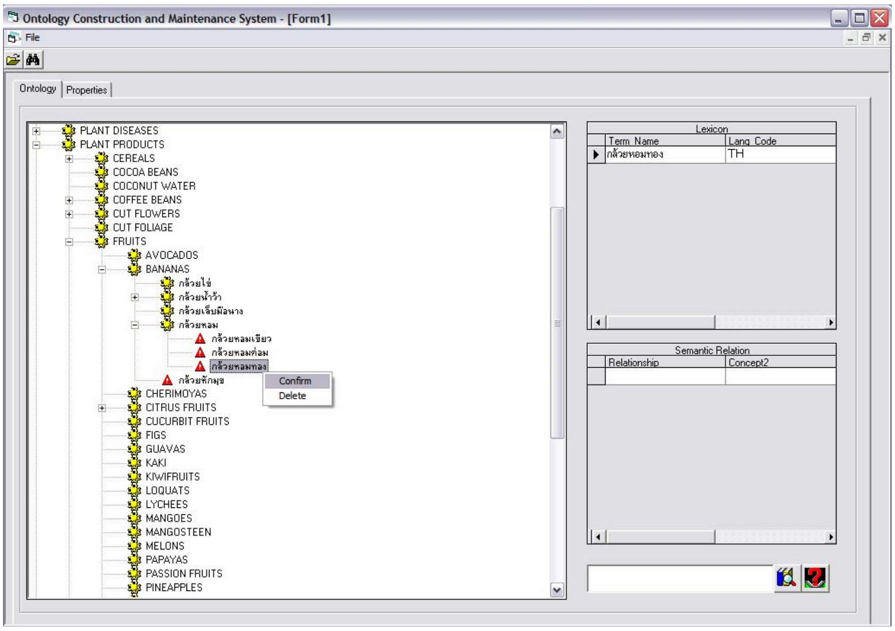
c) Merging

Figure 36 Operations for ontology reorganization

5. Verification Tool

Verification is required in order to ensure a high quality system, able to guide the expert to maintain the existing Ontology. This is why we have developed a user interface, allowing the expert to verify the quality of the output and to add related word pairs to the Ontological tree. Moreover, expert can delete node from the ontology tree if it is incorrect relationship.

Figure 37 shows the interface of the verification tool. The taxonomic relationships of the ontology are represented in the tree structure. When the node is selected, the tables in the right hand side will show the details of nodes that are the terms of this node and the semantic relationships of the node. The yellow icon show that the node and the relationship is correct and the red icon represent that this relationship is not confirm by the expert then the user can verify this relationship with confirm or delete this node. Moreover, the user can search the node by enter the query word in the text box at the bottom right hand side of window.



Lexicon

	Term Name	Lang Code
▶	กล้วยหอมทอง	TH

Semantic Relation

	Relationship	Concept2

Figure 37 Ontology verification tool