A Cluster Analysis Approach to Learning a Semantic Hierarchy for Machine Translation

Takefumi Yamazaki* and **Michael Pazzani**
Department of Information and Computer Science,
University of California, Irvine, CA, 92717
{tyamazak,pazzani}@ics.uci.edu

Abstract

This paper presents a method for learning a semantic hierarchy. The semantic hierarchy is used to bias the learning of rules for translating Japanese verbs to English. The creation of a semantic hierarchy is a form of constructive induction, since the nodes in the hierarchy may be regarded as new predicates that enable more accurate translation rules to be learned. Our experimental results show that a semantic hierarchy learned by our method is useful in learning translation rules for Japanese verbs.

1 Introduction

Constructing and maintaining a knowledge-based system is a difficult and time consuming endeavor. The required effort would be reduced if machine learning techniques were available to extract knowledge from examples. ALT-J/E, which is an experimental Japanese-English translation system developed at Nippon Telegraph and Telephone Corporation (NTT), is one example of a large knowledge-based system in which solutions to the knowledge acquisition bottleneck are definitely needed.

ALT-J/E contains a huge collection of translation rules (currently over 10,000) and a large semantic hierarchy (about 3,000 nodes). Each of the translation rules associates a Japanese sentence pattern with an appropriate English pattern. Each rule is expressed in terms of "semantic categories" which are nodes of the semantic hierarchy. To translate a Japanese sentence into English, ALT-J/E looks for the rule whose Japanese pattern matches the sentence best, and then uses the English pattern of that rule for translation.

Creating new translation rules as well as refining existing ones has proven to be extremely difficult and time-consuming because these tasks require considering the huge space of possible combinations of nodes in the hierarchy.

In a previous research, we have investigated the application of inductive learning techniques to the acquisition of translation rules [Almuallim et.al 1993, Almuallim et.al 1994]. Experimental results showed that this method was promising in facilitating the construction of rules. However, this method used a semantic hierarchy created by a human expert and did not consider learning this hierarchy. Learning the semantic hierarchy is essential and crucial for the machine translation task for the following reasons:

1. The quality of learned translation rules is determined by the quality of this hierarchy.
2. It is very difficult for a person to create a consistent hierarchy that does not include redundancy.
3. A different hierarchy may be necessary for different domains.

In this paper, we propose a method of learning a semantic hierarchy by analyzing translation rules learned without a hierarchy. The validity of a semantic hierarchy can be evaluated by learning translation rules, using the semantic hierarchy created by our method. In our implementation, FOCL [Pazzani & Kibler 1992] is used as a tool for learning translation rules from examples.

Constructing a semantic hierarchy can be regarded as a kind of constructive induction in that the general hierarchy terms serve as new predicates that can bias future learning. For example, without a hierarchy, a translation rule learned with a limited training set

*On leave from NTT Communication Science Laboratories, Address: 1-2356 Take, Yokosuka-Shi, Kanagawa-Ken, Japan 238-03, E-mail: yamazaki@nttkb.ntt.jp

might contain a rule that refers to a disjunction of the specific terms "pie" and "bread." However, with a semantic hierarchy, the learner would consider a more general term (e.g., "baked goods"). If the more general terms is used in a learned rule, the translation rule would apply to other types of baked goods (e.g., cakes).

Although we make no claims of psychological validity for our process of creating a semantic hierarchy, we do note that some psychologists (e.g., [Nelson 1977]) have claimed that semantic hierarchies of objects are acquired by children who notice that specific objects can be substituted for each other in certain contexts. Here, we consider that two objects are similar if the same English verb applies to them. For example, all instances of baked goods can be the objects of verbs such as "bake," "slice," and "knead," and the subjects of verbs such as "rise." Hierarchical relationships arise, since baked goods can be the object of verbs such as "eat," "cook," and "purchase" along with other categories such as "meat" and "vegetables" etc.

In the remainder of this paper, we give a brief description of the machine translation task. We then show how FOCL represents translation rules learned without a hierarchy. Finally, we propose our method for creating a semantic hierarchy and describe some preliminary experiments with this method.

2 Learning Tasks

We describe an overview of the machine translation task by using an experimental Japanese-English translation system, ALT-J/E system [Ikehara et al. 1989, Ikehara et al. 1990, Ikehara et al. 1991] as an example. In this work, we are concerned with the following components of the ALT-J/E system: The Semantic Hierarchy, Noun Dictionary, and the Translation Rules.

As shown in Figure 1, the **Semantic Hierarchy** is a sort of concept thesaurus represented as a tree structure in which each node is called a *semantic category*, or a *category* for simplicity. Edges in this structure represent "is-a" relations among the categories. For example, "Agents" and "Person" (see Figure 1) are both categories. The edge between these two categories indicates that any instance of "Person" is also an instance of "Agents." The current version of ALT-J/E's Semantic Hierarchy is 12 levels deep and has about 3000 nodes.

The **Noun Dictionary** maps each Japanese noun to its appropriate semantic category in the hierarchy. For example, the Noun Dictionary states that the noun "enpitsu" which means "pencil" in English, is an instance of the category "Stationery."

The **Translation Rules** in ALT-J/E associate Japanese patterns with English patterns. Currently, ALT-J/E uses roughly 10,000 of these rules. As Figure 2 shows, each translation rule has a Japanese pattern as its left-hand side and an English pattern as its right-hand side. For example, the first rule in this figure basically says that if the Japanese verb in a sentence is "YAKU," the category of its subject is "Person," and the category of its object is "Bread" or "Cake", then the following English pattern is to be used:

Subject "bake" Object.

Note that in this case the Japanese verb "YAKU" is translated into the English verb "bake." This same Japanese verb can also be translated into the English verbs "roast," "broil," "cremate" or "burn" depending on the context. These cases are handled by the four other rules given in Figure 2.

3 Learning Translation Rules with FOCL

In this section, we describe our task of learning translation rules[1] from examples. Each learning session concentrates on one Japanese verb, where it is desired to construct rules for mapping the verb in a given context to the appropriate English verb.

Each training example is represented as a pair of the following form

$$\langle [N_1 \equiv a, N_2 \equiv b, \cdots, N_n \equiv c],\ \text{E-Verb}\rangle$$

where each N_i represents a component of the sentence (subject, object, etc.), and each a, b, c is a category which is a node of a Semantic Hierarchy.

For instance, the sentence "itinen-sei ga enpitsu wo YAKU," which means "A freshman burns a pencil," gives the training example:

$$\langle [N_1 \equiv \text{"Person"}, N_2 \equiv \text{"Stationery"}],\ \text{E-Verb} = \text{"Burn"}\rangle$$

The job of a learning algorithm is to find the appropriate conditions that should hold in the context in order to map Japanese verbs into English verbs given a set of such examples.

FOCL represents translation rules as Horn Clauses. For example,

[1] To be strict, the present work is to learn what we call "partial translation rules". A partial translation rule consists of the left-hand side, along with the English verb of the right-hand side of a translation rule. In other words, the only difference between a translation rule and a partial translation rule is that the latter has only an English verb rather than a full English pattern as its right-hand side.

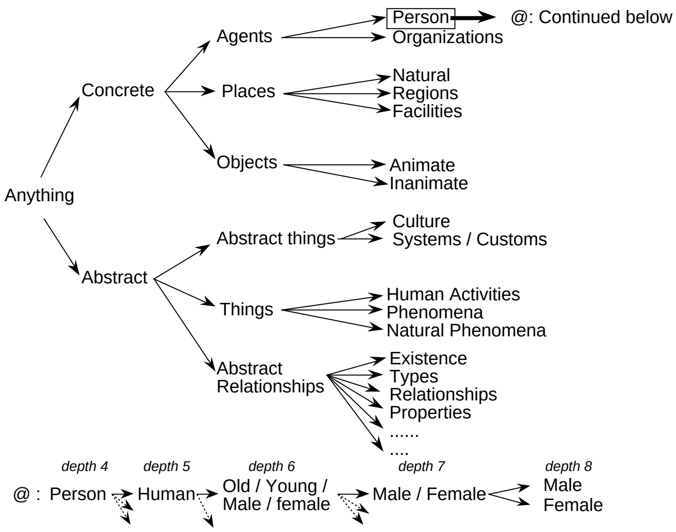

Figure 1: The upper levels of the Semantic Hierarchy in ALT-J/E.

yaku(Subject, Object, Everb) :-
isa(Subject, person), isa(Object, fish),
Everb = broil.

indicates that "yaku" is translated to "broil" when the object is a kind of fish and the subject is a person.

We have also extended the semantics of the isa predicate to allow a form of internal disjunction. Literals of the form isa(V, $\{c_1, ..., c_n\}$) are true if any of isa(V, c_1), ..., or isa(V, c_n) are true. FOCL has been modified to create such literals in a greedy, set covering manner. Using literals of the form isa(V, $\{c_1, ..., c_n\}$) is useful for representing disjunction of nodes in the existing hierarchy (e.g., isa(Object, {fish, seafood}). In addition, it is useful when there is no hierarchy. In this case, isa represents disjunctions of specific terms (e.g., isa(Object, {salmon, swordfish, lobster, shrimp}).

4 Learning a Semantic Hierarchy with a clustering method

In this section, we describe how a semantic hierarchy is created by analyzing the translation rules created by FOCL when learning without a semantic hierarchy. The frequency with which two terms occur together in an internal disjunction (i.e., in a literal of the form isa(V, $\{c_1, ..., c_n\}$) is used as a measure of the similarity between the terms. In particular, the mutual information gain ratio is used as follows. Let $p(c_i)$ be the probability that the term c_i appears in an internal disjunction of all learned translation rules. This is estimated by dividing the number of times that c_i appears in an internal disjunction by the total number of internal disjunctions. The probability that both c_i and c_j appear in an internal disjunction ($p(c_i \& c_j)$) is estimated in the same manner. The mutual information gain measure $log(p(c_i \& c_j)/p(c_i)p(c_j))$ is used as a measure of the similarity between two terms, c_i and c_j. This metric is often utilized in the natural language processing studies[Resnik 1990].

A matrix is created to store the value of the mutual information gain measure[2] for all terms.[3] Then, a standard statistical clustering algorithm, the average-linkage method[Aldenderfer & Blashfield 1984] is used to create a hierarchy. The procedure of creating a hierarchy by this method is described as follows:

1. It searches the dissimilarity matrix to find a pair of categories which has the highest value.
2. It combines this pair to create a new node.
3. The similarity between this node and the rest of categories is calculated by computing an average of the similarity between each of the categories, which creates the node, and others.
4. It modifies the matrix by adding the new node, deleting the categories used for creating the node, and assigning the similarity values, which are

[2] If $p(c_i \& c_j) = 0$, we use 0 as value in the matrix.

[3] In the statistical literature, such a matrix is called a dissimilarity matrix, since a value of 0 indicates that two terms are unrelated, and large values indicate a strong relationship between two terms. A similarity matrix is one in which 0 indicates that two items are identical and larger values indicate a difference between the two items. A similarity matrix might be formed by taking the Euclidean distance between two feature vectors.

IF			THEN		
	J-Verb	= "yaku"		Subj	= N_1
	N_1 (Subj)	≡ "Person"		E-Verb	="bake"
	N_2 (Obj)	≡ "Bread" or "Cake"		Obj	= N_2
IF			THEN		
	J-Verb	= "yaku"		Subj	= N_1
	N_1 (Subj)	≡ "Person"		E-Verb	="roast"
	N_2 (Obj)	≡ "Meat"		Obj	= N_2
IF			THEN		
	J-Verb	= "yaku"		Subj	= N_1
	N_1 (Subj)	≡ "Person"		E-Verb	="broil"
	N_2 (Obj)	≡ "Fish" or "Seafood"		Obj	= N_2
IF			THEN		
	J-Verb	= "yaku"		Subj	= N_1
	N_1 (Subj)	≡ "Agents"		E-Verb	="cremate"
	N_2 (Obj)	≡ "Person" or "Animals"		Obj	= N_2
IF			THEN		
	J-Verb	= "yaku"		Subj	= N_1
	N_1 (Subj)	≡ "Agents" or "Machines"		E-Verb	="burn"
	N_2 (Obj)	≡ "Places" or "Objects" or "Locations"		Obj	= N_2

Figure 2: Translation rules for the Japanese verb "YAKU". These rules are composed manually by human experts. " ≡ " indicates "an instance of".

computed by the previous step, to the new node.

5. It goes back to step 1.

This procedure repeats until all categories are combined into one.

This method is normally used to create binary hierarchies. We modified the method slightly to create hierarchies in which a node may have any number of children, provided all pairs of the children have the exact same value of the mutual information gain measure.

5 Experimental Results

The goal of the experiments reported here is to evaluate the quality of the semantic hierarchy generated with the proposed methods we have just described. In these experiments, training and testing examples were generated by using the translation rules and the semantic hierarchy created by human experts that are currently used in the ALT-J/E system. Namely, first, it searches the hierarchy to find all leaf nodes of categories which are used in the rule for each variable such as N_1 and N_2. Then, it choose one leaf node among them randomly as a value for the variable. We generated 400 examples for each English verb's rule. In this experiment, we deal with 24 different Japanese verbs, which correspond to about 100 English verbs. A total of approximately 40,000 examples are generated.

In order to learn translation rules for generating a hierarchy, we first ran FOCL on 24 Japanese verbs without a hierarchy by using 300 among 400 examples for each English verb's rule. A hierarchy was automatically created by our proposed method described in the previous section. Then, we used FOCL again to test the accuracy of translation rules by using the remaining 100 examples. This test was performed by 10-fold cross validation.

We ran experiments with the following three conditions:

1. Using FOCL to learn rules without any semantic hierarchy.
2. Using FOCL to learn rules with a semantic hierarchy created by a human expert.
3. Using FOCL to learn rules with a semantic hierarchy created by the clustering method.

Comparing Condition 2 with Condition 1 tells us the effectiveness of using a large semantic hierarchy on learning translation rules. Comparing Condition 3 with Condition 1 tells us how effectively the hierarchy generated by our methods works.

In order to evaluate Condition1 and Condition2, we used 487 examples to perform 5-fold cross-validation for each E-verb. Namely, 390 examples were used as training examples on each iteration. This number is the same number of training examples used in Condition 3 in which 300 examples for each E-verb were used for creating a hierarchy and 90 examples were used for learning rules in each iteration of the cross-validation.

Table 1 shows a list of these verbs, along with the accuracy levels obtained by each method. In the table, "Without Hierarchy," "Hierarchy by human," and "Generated Hierarchy" denote Condition 1, Condition 2, and Condition 3, respectively.

This result shows that the accuracy for many of the Japanese verbs improves with the semantic hierarchy generated by our methods compared to the method without using any hierarchy. This means that the generated hierarchy is effective in learning translation rules.

However, the human-created hierarchy results in more accurate rules than the learned hierarchy. Only 90 examples were used to learn the rules after learning the hierarchy, because 300 examples were used to learn the hierarchy. In addition, a diverse group of verbs was used, and there may not be many opportunities for sharing newly created hierarchical nodes among the verbs. We intend on running experiments with an assortment of more similar verbs to see if we get more transfer.

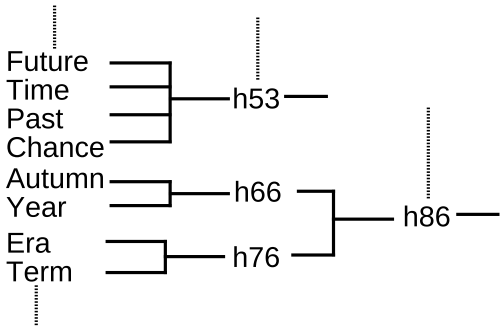

Figure 3: Part of the Generated Semantic Hierarchy

nobiru(Subject, Until, Everb) :-
isa(Until, {h53,h86}), Everb = be-put-off.
nobiru(Subject, Until, Everb) :-
isa(Subject, seed), Everb = spread.
nobiru(Subject, Until, Everb) :-
isa(Until, age), Everb = be-prolonged.

Figure 4: Part of the Learned rule for the Japanese verb "Nobiru"

Figure 3 shows a part of the generated semantic hierarchy (where "Hxx" indicates new nodes). This hierarchy reproduces a part of the hierarchy created manually. This suggests that semantically reasonable nodes are created by our method.

The rules learned with the generated semantic hierarchy are expressed with new nodes and their lengths are more compact than those of rules generated without using any hierarchy. Figure 4 shows a part of the learned rules for the Japanese verb "Nobiru." You can find new nodes shown in Figure 3 in this rule.

6 Conclusion

This paper reported our work toward the acquisition of a semantic hierarchy and Japanese-English translation rules through the use of clustering techniques and an inductive learning tool, FOCL. Although we use FOCL in our implementation as a tool for learning rules, the methods developed in this paper are more general and can easily be adapted within other learners. For example, an inductive algorithm for learning internal disjunctive expressions[Haussler 1988] could be used instead of FOCL. Namely, our method can be applied by using the information about categories appeared in internal disjunctions of the rules learned by this Haussler's algorithm. Another inductive methods such as decision tree algorithms[Quinlan 1986] may also be exploited as an inductive tool if they are modified so that each node of the tree can represented by internal disjunctive expressions.

We employed a clustering method with mutual information metric for learning a semantic hierarchy. Experimental results suggest that the approach we have chosen to build a semantic hierarchy is promising for our task. This learning task is attractive because it has a clear measure to evaluate the results of constructive induction by demonstrating that the newly created predicates facilitate learning of other concepts.

We presented here a method of creating a hierarchy from scratch. As a next step, it is very important to consider the method of modifying the current hierarchy created manually. As we stated in the introduction, the current hierarchy may include redundancy though it is beneficial so far. We hope to show experimentally that revising the human-created hierarchy results in learning more accurate translation rules.

References

[Aldenderfer & Blashfield 1984] Aldenderfer, M., and Blashfield, R., "Cluster Analysis", *SAGE University Papers*, 1984.

[Almuallim et.al 1993] Almuallim, H., Yamazaki T., AKiba, Y., Yokoo, A., Kaneda, S., and Kawaoka,

Table 1: Accuracy Result : 3 methods : 24 Japanese verbs

Japanese Verb	*English Verbs*	*Accuracy %* Without Hierarchy	Hierarchy (human)	Generated Hierarchy
Yaku	burn, bake, roast, broil, cremate	68	93	91
Nomu	drink, take, accept, eat	99	100	100
Okonau	play, hold, conduct	98	99	94
Oujiru	enter, answer, meet	98	100	100
Toku	undo, dispel, solve	99	99	99
Tsukau	use, spend, employ	63	93	91
Ataru	hit,prove, win, work, fall, ...	54	91	71
Atsumeru	gather, collect, recruit, ...	66	99	87
Eru	get, gain, win	99	98	99
Karamu	pick, twine, be involved, ...	77	100	89
Kotaeru	meet, affect, respond, ...	98	100	99
Nobiru	spread, be extended, be put off,..	60	100	99
Noru	get-on, ride, take, yield, ...	55	96	83
Nuru	apply, spread, color, coat	100	99	99
Osamaru	be-at-peace, calm-down, ...	72	100	83
Oshimu	be-sorry, lament, spare,...	71	96	79
Sasageru	hold, offer, sacrifice, ...	78	99	97
Soeru	add, attach, garnish, lend	90	99	93
Taku	heat, burn, heat, light, use	100	100	100
Taosu	upset, defeat, overthrow, blow	57	92	83
Tataku	beat, knock, clap, pat	46	89	87
Umu	bear, lay, spawn, produce,...	87	99	96
Yaburu	tear, violate, beat, break	64	98	92
Yashinau	support, train, raise, cultivate	43	97	74

T., "Acquisition of Machine Translation Rules Using Inductive Learning Techniques", *IJCAI-93 Workshop "Machine Learning and Knowledge Acquisition*, 1993.

[Almuallim et.al 1994] Almuallim, H., Akiba, Y., Yamazaki T., Yokoo, A., and Kaneda, S., "A tool for the Acquisition of Japanese-English Machine Translation Rules Using Inductive Learning Techniques", *The 10th Conference of Artificial Intelligence for Applications*, 1994.

[Haussler 1988] Haussler, D., "Quantifying inductive bias: AI learning algorithms and Valiant's learning framework", *Artificial Intelligence,* 26(2), 177–221, 1988.

[Ikehara et al. 1989] Ikehara, S., Miyazaki, M., Shirai, S. and Yokoo, A., "An Approach to Machine Translation Method based on Constructive Process Theory", *Review of ECL,* vol.37, No.1, 39–44, 1989.

[Ikehara et al. 1990] Ikehara, S., Shirai, S., Yokoo, A. and Nakaiwa, H., "Toward an MT System without Pre-Editing—Effects of New Methods in ALT-J/E", *Proc. of MT Summit-3*, 1990.

[Ikehara et al. 1991] Ikehara, S., Yokoo, A. and Miyazaki, M., "Semantic Analysis Dictionaries for Machine Translation", (in Japanese), *IEICE Report, NLC 91-19, Institute of Electronics, Information and Communication Engineers, Japan*, 1991.

[Nelson 1977] Nelson, K., "Cognitive development and the acquisition of concepts", R. C. Anderson, R. J. Spiro, & W. Montague (Eds.), *Schooling and the acquisition of knowledge*, Hillsdale, NJ: Lawrence Erlbaum.

[Pazzani & Kibler 1992] Pazzani, M., and Kibler, D., "The utility of knowledge in inductive learning", *Machine Learning*, 9(1):57-94, 1992.

[Quinlan 1986] Quinlan, J. R. "Induction of Decision Trees", *Machine Learning,* 1 (1), 1986.

[Quinlan 1990] Quinlan, J. R., "Learning logical definitions from relations", *Machine Learning,* 5 (3), 1990.

[Resnik 1990] Resnik, P., "WordNet and Distributional Analysis: A Class-based Approach to Lexical Discovery", *AAAI-92 workshop "Statistically-Based Natural Language Programming Techniques"*, 1992.