General-to-Specific Model Selection for Subcategorization Preference*

Takehito Utsuro and **Takashi Miyata** and **Yuji Matsumoto**
Graduate School of Information Science, Nara Institute of Science and Technology
8916-5, Takayama-cho, Ikoma-shi, Nara, 630-0101, JAPAN
E-mail: utsuro@is.aist-nara.ac.jp, URL: http://cl.aist-nara.ac.jp/~utsuro/

Abstract

This paper proposes a novel method for learning probability models of subcategorization preference of verbs. We consider the issues of *case dependencies* and *noun class generalization* in a uniform way by employing the maximum entropy modeling method. We also propose a new model selection algorithm which starts from the most *general* model and gradually examines more *specific* models. In the experimental evaluation, it is shown that both of the case dependencies and specific sense restriction selected by the proposed method contribute to improving the performance in subcategorization preference resolution.

1 Introduction

In empirical approaches to parsing, lexical/semantic collocation extracted from corpus has been proved to be quite useful for ranking parses in syntactic analysis. For example, Magerman (1995), Collins (1996), and Charniak (1997) proposed statistical parsing models which incorporated lexical/semantic information. In their models, syntactic and lexical/semantic features are dependent on each other and are combined together. This paper also proposes a method of utilizing lexical/semantic features for the purpose of applying them to ranking parses in syntactic analysis. However, unlike the models of Magerman (1995), Collins (1996), and Charniak (1997), we assume that syntactic and lexical/semantic features are independent. Then, we focus on extracting lexical/semantic collocational knowledge of verbs which is useful in syntactic analysis.

More specifically, we propose a novel method for learning a probability model of subcategorization preference of verbs. In general, when learning lexical/semantic collocational knowledge of verbs from corpus, it is necessary to consider the two issues of 1) *case dependencies*, and 2) *noun class generalization*. When considering 1), we have to decide which cases are dependent on each other and which cases are optional and independent of other cases. When considering 2), we have to decide which superordinate class generates each observed leaf class in the verb-noun collocation. So far, there exist several works which worked on these two issues in learning collocational knowledge of verbs and also evaluated the results in terms of syntactic disambiguation. Resnik (1993) and Li and Abe (1995) studied how to find an optimal abstraction level of an argument noun in a tree-structured thesaurus. Their works are limited to only one argument. Li and Abe (1996) also studied a method for learning dependencies between case slots and reported that dependencies were discovered only at the slot-level and not at the class-level.

Compared with these previous works, this paper proposes to consider the above two issues in a uniform way. First, we introduce a model of generating a collocation of a verb and argument/adjunct nouns (section 2) and then view the model as a probability model (section 3). As a model learning method, we adopt the maximum entropy model learning method (Della Pietra et al., 1997; Berger et al., 1996). *Case dependencies* and *noun class generalization* are represented as *features* in the maximum entropy approach. *Features* are allowed to have overlap and this is quite advantageous when we consider case dependencies and noun class generalization in parameter estimation. An *optimal* model is selected by searching for an optimal set of features, i.e, optimal case dependencies and optimal noun class generalization levels. As the feature selection process, this paper proposes a new feature selection algorithm which starts from the most *general* model and gradually examines more *specific* models (section 4). As the model evaluation criterion during the model search from general to specific ones, we employ the description length of the model and guide the search process so as to minimize the description length (Rissanen, 1984). Then, after obtaining a sequence of subcategorization preference models which are totally ordered from general to specific, we select an approximately optimal subcategorization preference model according to the accuracy of subcategorization preference test. In the experimental evaluation of performance of subcatego-

* This research was partially supported by the Ministry of Education, Science, Sports and Culture, Japan, Grant-in-Aid for Encouragement of Young Scientists, 09780338, 1998. An extended version of this paper is available from the above URL.

rization preference, it is shown that both of the case dependencies and specific sense restriction selected by the proposed method contribute to improving the performance in subcategorization preference resolution (section 5).

2 A Model of Generating a Verb-Noun Collocation from Subcategorization Frame(s)

This section introduces a model of generating a verb-noun collocation from subcategorization frame(s).

2.1 Data Structure

Verb-Noun Collocation *Verb-noun collocation* is a data structure for the collocation of a verb and all of its argument/adjunct nouns. A verb-noun collocation e is represented by a feature structure which consists of the verb v and all the pairs of co-occurring case-markers p and thesaurus classes c of case-marked nouns:

$$e = \begin{bmatrix} pred : v \\ p_1 : c_1 \\ \vdots \\ p_k : c_k \end{bmatrix} \quad (1)$$

We assume that a *thesaurus* is a tree-structured type hierarchy in which each node represents a semantic class, and each thesaurus class $c_1, \ldots, c_k$ in a verb-noun collocation is a leaf class in the thesaurus. We also introduce $\preceq_c$ as the superordinate-subordinate relation of classes in a thesaurus: $c_1 \preceq_c c_2$ means that c_1 is subordinate to c_2. [1]

Subcategorization Frame A *subcategorization frame* s is represented by a feature structure which consists of a verb v and the pairs of case-markers p and sense restriction c of case-marked argument/adjunct nouns:

$$s = \begin{bmatrix} pred : v \\ p_1 : c_1 \\ \vdots \\ p_l : c_l \end{bmatrix} \quad (2)$$

Sense restriction $c_1, \ldots, c_l$ of case-marked argument/adjunct nouns are represented by classes at arbitrary levels of the thesaurus.

Subsumption Relation We introduce the *subsumption relation* $\preceq_{sf}$ of a *verb-noun collocation* e and a *subcategorization frame* s:

$e \preceq_{sf} s$ iff. for each case-marker p_i in s and its noun class c_{si}, there exists the same case-marker p_i in e and its noun class c_{ei} is subordinate to c_{si}, i.e. $c_{ei} \preceq_c c_{si}$

The subsumption relation $\preceq_{sf}$ is applicable also as a subsumption relation of two subcategorization frames.

2.2 Generating a Verb-Noun Collocation from Subcategorization Frame(s)

Suppose a verb-noun collocation e is given as:

$$e = \begin{bmatrix} pred : v \\ p_1 : c_{e1} \\ \vdots \\ p_k : c_{ek} \end{bmatrix} \quad (3)$$

Then, let us consider a tuple $\langle s_1, \ldots, s_n \rangle$ of *partial subcategorization frames* which satisfies the following requirements: i) the unification $s_1 \wedge \cdots \wedge s_n$ of all the partial subcategorization frames has exactly the same case-markers as e has as in (4), ii) each semantic class c_{si} of a case-marked noun of the partial subcategorization frames is superordinate to the corresponding leaf semantic class c_{ei} of e as in (5), and iii) any pair s_i and $s_{i'}$ $(i \neq i')$ do not have common case-markers as in (6):

$$s_1 \wedge \cdots \wedge s_n = \begin{bmatrix} pred : v \\ p_1 : c_{s1} \\ \vdots \\ p_k : c_{sk} \end{bmatrix} \quad (4)$$

$$c_{ei} \preceq_c c_{si} \quad (i=1,\ldots,k) \quad (5)$$

$$s_i = \begin{bmatrix} pred : v \\ \vdots \\ p_{ij} : c_{ij} \\ \vdots \end{bmatrix}, \quad \begin{array}{l} \forall j \forall j' \; p_{ij} \neq p_{i'j'} \\ (i,i'=1,\ldots,n, \; i \neq i') \end{array} \quad (6)$$

When a tuple $\langle s_1, \ldots, s_n \rangle$ satisfies the above three requirements, we assume that the tuple $\langle s_1, \ldots, s_n \rangle$ can *generate* the verb-noun collocation e and denote as below:

$$\langle s_1, \ldots, s_n \rangle \longrightarrow e \quad (7)$$

As we will describe in section 3.2, we assume that the partial subcategorization frames $s_1, \ldots, s_n$ are regarded as events occurring *independently* of each other and each of them is assigned an independent parameter.

2.3 Example

This section shows how we can incorporate *case dependencies* and *noun class generalization* into the model of generating a verb-noun collocation from a tuple of partial subcategorization frames.

[1] Although we ignore sense ambiguities of case-marked nouns in the definitions of this section, in the current implementation, we deal with sense ambiguities of case-marked nouns by deciding that a class c is superordinate to an ambiguous leaf class C_l if c is superordinate to at least one of the possible unambiguous classes of C_l.

The Ambiguity of Case Dependencies The problem of the ambiguity of case dependencies is caused by the fact that, only by observing each verb-noun collocation in corpus, it is not decidable which cases are dependent on each other and which cases are optional and independent of other cases. Consider the following example:

Example 1

Kodomo-ga kouen-de juusu-wo nomu.
child-NOM park-at juice-ACC drink
(A child drinks juice at the park.)

The verb-noun collocation is represented as a feature structure e below:

$$e = \begin{bmatrix} pred : nomu \\ ga : c_c \\ wo : c_j \\ de : c_p \end{bmatrix} \quad (8)$$

where c_c, c_p, and c_j represent the leaf classes (in the thesaurus) of the nouns *"kodomo(child)"*, *"kouen(park)"*, and *"juusu(juice)"*.

Next, we assume that the concepts *"human"*, *"place"*, and *"beverage"* are superordinate to *"kodomo(child)"*, *"kouen(park)"*, and *"juusu(juice)"*, respectively, and introduce the corresponding classes c_{hum}, c_{plc}, and c_{bev} as sense restriction in subcategorization frames. Then, according to the dependencies of cases, we can consider several patterns of subcategorization frames each of which can generate the verb-noun collocation e.

If the three cases *"ga(NOM)"*, *"wo(ACC)"*, and *"de(at)"* are dependent on each other and it is not possible to find any division into several independent subcategorization frames, e can be regarded as generated from a subcategorization frame containing all of the three cases:

$$\left\langle \begin{bmatrix} pred : nomu \\ ga : c_{hum} \\ wo : c_{bev} \\ de : c_{plc} \end{bmatrix} \right\rangle \longrightarrow e \quad (9)$$

Otherwise, if only the two cases *"ga(NOM)"* and *"wo(ACC)"* are dependent on each other and the *"de(at)"* case is independent of those two cases, e can be regarded as generated from the following two subcategorization frames independently:

$$\left\langle \begin{bmatrix} pred : nomu \\ ga : c_{hum} \\ wo : c_{bev} \end{bmatrix}, \begin{bmatrix} pred : nomu \\ de : c_{plc} \end{bmatrix} \right\rangle \longrightarrow e \quad (10)$$

The Ambiguity of Noun Class Generalization The problem of the ambiguity of noun class generalization is caused by the fact that, only by observing each verb-noun collocation in corpus, it is not decidable which superordinate class generates each observed leaf class in the verb-noun collocation. Let us again consider Example 1. We assume that the concepts *"mammal"* and *"liquid"* are superordinate to *"human"* and *"beverage"*, respectively, and introduce the corresponding classes c_{mam} and c_{liq}. If we additionally allow these superordinate classes as sense restriction in subcategorization frames, we can consider several additional patterns of subcategorization frames which can generate the verb-noun collocation e.

Suppose that only the two cases *"ga(NOM)"* and *"wo(ACC)"* are dependent on each other and the *"de(at)"* case is independent of those two cases as in the formula (10). Since the leaf class c_c (*"child"*) can be generated from either c_{hum} or c_{mam}, and also the leaf class c_j (*"juice"*) can be generated from either c_{bev} or c_{liq}, e can be regarded as generated according to either of the four formulas (10) and (11)∼(13):

$$\left\langle \begin{bmatrix} pred : nomu \\ ga : c_{mam} \\ wo : c_{bev} \end{bmatrix}, \begin{bmatrix} pred : nomu \\ de : c_{plc} \end{bmatrix} \right\rangle \longrightarrow e \quad (11)$$

$$\left\langle \begin{bmatrix} pred : nomu \\ ga : c_{hum} \\ wo : c_{liq} \end{bmatrix}, \begin{bmatrix} pred : nomu \\ de : c_{plc} \end{bmatrix} \right\rangle \longrightarrow e \quad (12)$$

$$\left\langle \begin{bmatrix} pred : nomu \\ ga : c_{mam} \\ wo : c_{liq} \end{bmatrix}, \begin{bmatrix} pred : nomu \\ de : c_{plc} \end{bmatrix} \right\rangle \longrightarrow e \quad (13)$$

3 Maximum Entropy Modeling of Subcategorization Preference

This section describes how we apply the maximum entropy modeling approach of Della Pietra et al. (1997) and Berger et al. (1996) to model learning of subcategorization preference.

3.1 Maximum Entropy Modeling

Given the training sample $\mathcal{E}$ of the events (x, y), our task is to estimate the conditional probability $p(y \mid x)$ that, given a context x, the process will output y. In order to express certain features of the whole event (x, y), a binary-valued indicator function is introduced and called a *feature function*. Usually, we suppose that there exists a large collection $\mathcal{F}$ of candidate features, and include in the model only a subset $\mathcal{S}$ of the full set of candidate features $\mathcal{F}$. We call $\mathcal{S}$ the set of *active* features. Now, we assume that $\mathcal{S}$ contains n feature functions. For each feature $f_i (\in \mathcal{S})$, the sets V_{xi} and V_{yi} indicate the sets of the values of x and y for that feature. According to those sets, each feature function f_i will be defined as follows:

$$f_i(x, y) = \begin{cases} 1 & \text{if } x \in V_{xi} \text{ and } y \in V_{yi} \\ 0 & \text{otherwise} \end{cases}$$

Then, in the maximum entropy modeling approach, the model with the maximum entropy is selected among the possible models. With this constraint, the conditional probability of the output y given the context x can be estimated as the following $p_\lambda(y \mid x)$ of the form of the exponential family, where a *parameter* λ_i is introduced

for each feature f_i.

$$p_\lambda(y \mid x) = \frac{\exp\left(\sum_i \lambda_i f_i(x,y)\right)}{\sum_y \exp\left(\sum_i \lambda_i f_i(x,y)\right)} \quad (14)$$

The parameter values λ^*_i are estimated by an algorithm called *Improved Iterative Scaling* (IIS) algorithm.

Feature Selection by One-by-one Feature Adding The feature selection process presented in Della Pietra et al. (1997) and Berger et al. (1996) is an incremental procedure that builds up $\mathcal{S}$ by successively adding features one-by-one. It starts with $\mathcal{S}$ as empty, and, at each step, selects the candidate feature which, when adjoined to the set of active features $\mathcal{S}$, produces the greatest increase in log-likelihood of the training sample.

3.2 Modeling Subcategorization Preference

Events In our task of model learning of subcategorization preference, each *event* (x, y) in the training sample is a verb-noun collocation e, which is defined in the formula (1). A verb-noun collocation e can be divided into two parts: one is the verbal part e_v containing the verb v while the other is the nominal part e_p containing all the pairs of case-markers p and thesaurus leaf classes c of case-marked nouns:

$$e = e_v \wedge e_p = [\ pred : v\ ] \wedge \begin{bmatrix} p_1 : c_1 \\ \vdots \\ p_k : c_k \end{bmatrix}$$

Then, we define the *context* x of an event (x, y) as the verb v and the *output* y as the nominal part e_p of e, and each event in the training sample is denoted as (v, e_p):

$$x \equiv v, \quad y \equiv e_p$$

Features We represent each partial subcategorization frame as a *feature* in the maximum entropy modeling. According to the possible variations of case dependencies and noun class generalization, we consider every possible patterns of subcategorization frames which can generate a verb-noun collocation, and then construct the full set $\mathcal{F}$ of candidate features. Next, for the given verb-noun collocation e, tuples of partial subcategorization frames which can generate e are collected into the set $SF(e)$ as below:

$$SF(e) = \left\{ \langle s_1, \ldots, s_n \rangle \,\middle|\, \langle s_1, \ldots, s_n \rangle \longrightarrow e \right\}$$

Then, for each partial subcategorization frame s, a binary-valued feature function $f_s(v, e_p)$ is defined to be true if and only if at least one element of the set $SF(e)$ is a tuple $\langle s_1, \ldots, s, \ldots, s_n \rangle$ that contains s:

$$f_s(v, e_p) = \begin{cases} 1 & \text{if } \exists \langle s_1, \ldots, s, \ldots, s_n \rangle \\ & \quad \in SF(e = ([pred : v] \wedge e_p)) \\ 0 & \text{otherwise} \end{cases}$$

In the maximum entropy modeling approach, each feature is assigned an independent parameter, i.e., each (partial) subcategorization frame is assigned an independent parameter.

Parameter Estimation Suppose that the set $\mathcal{S}(\subseteq \mathcal{F})$ of active features is found by the procedure of the next section. Then, the parameters of subcategorization frames are estimated according to IIS Algorithm and the conditional probability distribution $p_{\mathcal{S}}(e_p \mid v)$ is given as:

$$p_{\mathcal{S}}(e_p \mid v) = \frac{\exp\left(\sum_{f_i \in \mathcal{S}} \lambda_i f_i(v, e_p)\right)}{\sum_{e_p} \exp\left(\sum_{f_i \in \mathcal{S}} \lambda_i f_i(v, e_p)\right)} \quad (15)$$

4 General-to-Specific Feature Selection

This section describes the new feature selection algorithm which utilizes the subsumption relation of subcategorization frames. It starts from the most *general* model, i.e., a model with no case dependency as well as the most general sense restrictions which correspond to the highest classes in the thesaurus. This starting model has high coverage of the test data. Then, the algorithm gradually examines more *specific* models with case dependencies as well as more specific sense restrictions which correspond to lower classes in the thesaurus. The model search process is guided by a model evaluation criterion.

4.1 Partially-Ordered Feature Space

In section 2.1, we introduced subsumption relation $\preceq_{sf}$ of two subcategorization frames. All the subcategorization frames are partially ordered according to this subsumption relation, and elements of the set $\mathcal{F}$ of candidate features constitute a partially ordered feature space.

Constraint on Active Feature Set Throughout the feature selection process, we put the following constraint on the active feature set $\mathcal{S}$:

> *Case Covering Constraint*: for each verb-noun collocation in the training set $\mathcal{E}$, each case p (and the leaf class marked by p) of e has to be *covered* by at least one feature in $\mathcal{S}$.

Initial Active Feature Set Initial set $\mathcal{S}_0$ of active features is constructed by collecting features which are not subsumed by any other candidate features in $\mathcal{F}$:

$$\mathcal{S}_0 = \left\{ f_s \,\middle|\, \forall f_{s'} (\neq f_s) \in \mathcal{F},\ s \not\preceq_{sf} s' \right\} \quad (16)$$

This constraint on the initial active feature set means that each feature in $\mathcal{S}_0$ has only one case and the sense restriction of the case is (one of) the most general class(es).

Candidate Non-active Features for Replacement At each step of feature selection, one of the active features is replaced with several non-active features. Let $\mathcal{G}$ be a set of non-active features which have never been active until that step. Then, for each active feature $f_s(\in \mathcal{S})$, the set $D_{f_s}(\subseteq \mathcal{G})$ of candidate non-active features with which f_s is replaced has to satisfy the following two requirements[2] [3].

1. *Subsumption with s*: for each element $f_{s'}$ of D_{f_s}, s' has to be subsumed by s.
2. *Upper Bound of* $\mathcal{G}$: for each element $f_{s'}$ of D_{f_s}, and for each element f_t of $\mathcal{G}$, t does not subsume s', i.e., D_{f_s} is a subset of the upper bound of $\mathcal{G}$ with respect to the subsumption relation $\preceq_{sf}$.

Among all the possible replacements, the most appropriate one is selected according to a model evaluation criterion.

4.2 Model Evaluation Criterion

As the model evaluation criterion during feature selection, we consider the following two types.

4.2.1 MDL Principle

The MDL (Minimum Description Length) principle (Rissanen, 1984) is a model selection criterion. It is designed so as to "select the model that has as much fit to a given data as possible and that is as simple as possible." The MDL principle selects the model that minimizes the following *description length* $l(M, D)$ of the probability model M for the data D:

$$l(M,D) = -\log L_M(D) + \frac{1}{2} N_M \log |D| \quad (17)$$

where $\log L_M(D)$ is the log-likelihood of the model M to the data D, N_M is the number of the parameters in the model M, and $|D|$ is the size of the data D.

Description Length of Subcategorization Preference Model The description length $l(p_S, \mathcal{E})$ of the probability model p_S (of (15)) for the training data set $\mathcal{E}$ is given as below:[4]

$$l(p_S,\mathcal{E}) = -\sum_{(v,e_p)\in\mathcal{E}} \log p_S(e_p \mid v) + \frac{1}{2}|\mathcal{S}| \log |\mathcal{E}| \quad (18)$$

[2]The *general-to-specific* feature selection considers only a small portion of the non-active features as the next candidate for the active feature, while the feature selection by *one-by-one feature adding* considers all the non-active features as the next candidate. Thus, in terms of efficiency, the *general-to-specific* feature selection has an advantage over the *one-by-one feature adding* algorithm, especially when the number of the candidate features is large.

[3]As long as the *case covering constraint* is satisfied, the set D_{f_s} of candidate non-active features with which f_s is replaced could be an empty set $\emptyset$.

[4]More precisely, we slightly modify the probability model p_S by multiplying the probability of generating the verb-noun collocation e from the (partial) subcategorization frames that correspond to active features evaluating to true for e, and then apply the MDL principle to this modified model. The probability of generating a verb-noun collocation from (partial) subcategorization frames is simply estimated as the product of the probabilities of generating each leaf-class in the verb-noun collocation from the corresponding superordinate class in the subcategorization frame. With this generation probability, the more general the sense restriction of the subcategorization frames is, the less fit the model has to the data, and the greater the data description length (the first term of (18)) of the model is. Thus, this modification causes the feature selection process to be more sensitive to the sense restriction of the model.

4.2.2 Subcategorization Preference Test using Positive/Negative Examples

The other type of the model evaluation criterion is the performance in the subcategorization preference test presented in Utsuro and Matsumoto (1997), in which the goodness of the model is measured according to how many of the positive examples can be judged as more appropriate than the negative examples. This subcategorization preference test can be regarded as modeling the subcategorization ambiguity of an argument noun in a Japanese sentence with more than one verbs like the one in Example 2.

Example 2

TV-de	mouketa	shounin-wo	mita
TV-by/on	*earn money*	*merchant-ACC*	*see*

(If the phrase *"TV-de"(by/on TV)* modifies the verb *"mouketa"(earn money)*, the sentence means that "(Somebody) saw a merchant who earned money by (selling) TV." On the other hand, if the phrase *"TV-de"(by/on TV)* modifies the verb *"mita"(see)*, the sentence means that "On TV, (somebody) saw a merchant who earned money.")

Negative examples are artificially generated from the positive examples by choosing a case element in a positive example of one verb at random and moving it to a positive example of another verb.

Compared with the calculation of the description length $l(p_S, \mathcal{E})$ in (18), the calculation of the accuracy of subcategorization preference test requires comparison of probability values for sufficient number of positive and negative data and its computational cost is much higher than that of calculating the description length. Therefore, at present, we employ the description length $l(p_S, \mathcal{E})$ in (18) as the model evaluation criterion during the general-to-specific feature selection procedure, which we will describe in the next section in detail. After obtaining a sequence of active feature sets (i.e., subcategorization preference models) which are totally ordered from general to specific, we select an optimal subcategorization preference model according to the accuracy of subcategorization preference test, as we will describe in section 4.4.

4.3 Feature Selection Algorithm

The following gives the details of the *general-to-specific* feature selection algorithm, where the de-

scription length $l(p_S, \mathcal{E})$ in (18) is employed as the model evaluation criterion:[5]

General-to-Specific Feature Selection

Input: Training data set $\mathcal{E}$;
collection $\mathcal{F}$ of candidate features
Output: Set $\mathcal{S}$ of active features;
model p_S incorporating these features

1. Start with $\mathcal{S} = \mathcal{S}_0$ of the definition (16) and with $\mathcal{G} = \mathcal{F} - \mathcal{S}_0$
2. Do for each active feature $f \in \mathcal{S}$ and every possible replacement $D_f \subseteq \mathcal{G}$:

   Compute the model $p_{\mathcal{S} \cup D_f - \{f\}}$ using IIS Algorithm.

   Compute the decrease in the description length of (18).
3. Check the termination condition[6]
4. Select the feature $\hat{f}$ and its replacement $D_{\hat{f}}$ with maximum decrease in the description length
5. $\mathcal{S} \longleftarrow \mathcal{S} \cup D_{\hat{f}} - \{\hat{f}\}, \quad \mathcal{G} \longleftarrow \mathcal{G} - D_{\hat{f}}$
6. Compute p_S using IIS Algorithm
7. Go to step 2

4.4 Selecting a Model with Approximately Optimal Subcategorization Preference Accuracy

Suppose that we are constructing subcategorization preference models for the verbs $v_1, \ldots, v_m$. By the general-to-specific feature selection algorithm in the previous section, for each verb v_i, a totally ordered sequence of n_i active feature sets $\mathcal{S}_{i0}, \ldots, \mathcal{S}_{in_i}$ (i.e., subcategorization preference models) are obtained from the training sample $\mathcal{E}$. Then, using another training sample $\mathcal{E}'$ which is different from $\mathcal{E}$ and consists of positive as well as negative data, a model with optimal subcategorization preference accuracy is approximately selected by the following procedure. Let $\mathcal{T}_1, \ldots, \mathcal{T}_m$ denote the current sets of active features for verbs $v_1, \ldots, v_m$, respectively:

1. Initially, for each verb v_i, set $\mathcal{T}_i$ as the most general one $\mathcal{S}_{i0}$ of the sequence $\mathcal{S}_{i0}, \ldots, \mathcal{S}_{in_i}$.
2. For each verb v_i, from the sequence $\mathcal{S}_{i1}, \ldots, \mathcal{S}_{in_i}$, search for an active feature set which gives a maximum subcategorization preference accuracy for $\mathcal{E}'$, then set $\mathcal{T}_i$ as it.
3. Repeat the same procedure as 2.
4. Return the current sets $\mathcal{T}_1, \ldots, \mathcal{T}_m$ as the approximately optimal active feature sets $\hat{\mathcal{S}}_1, \ldots, \hat{\mathcal{S}}_m$ for verbs $v_1, \ldots, v_m$, respectively.

[5] Note that this feature selection algorithm is a hill-climbing one and the model selected here may have a description length greater than the *global minimum*.

[6] In the present implementation, the feature selection process is terminated after the description length of the model stops decreasing and then certain number of active features are replaced.

5 Experiment and Evaluation

5.1 Corpus and Thesaurus

As the training and test corpus, we used the EDR Japanese bracketed corpus (EDR, 1995), which contains about 210,000 sentences collected from newspaper and magazine articles. We used 'Bunrui Goi Hyou'(BGH) (NLRI, 1993) as the Japanese thesaurus. BGH has a seven-layered abstraction hierarchy and more than 60,000 words are assigned at the leaves and its nominal part contains about 45,000 words.

5.2 Training/Test Events and Features

We conduct the model learning experiment under the following conditions: i) the noun class generalization level of each feature is limited to above the level 5 from the root node in the thesaurus, ii) since verbs are independent of each other in our model learning framework, we collect verb-noun collocations of one verb into a training data set and conduct the model learning procedure for each verb separately.

For the experiment, seven Japanese verbs[7] are selected so that the difficulty of the subcategorization preference test is balanced among verb pairs. The number of training events for each verb varies from about 300 to 400, while the number of candidate features for each verb varies from 200 to 1,350. From this data, we construct the following three types of data set, each pair of which has no common element: i) the training data $\mathcal{E}$ which consists of positive data only, and is used for selecting a sequence of active feature sets by the general-to-specific feature selection algorithm in section 4.3, ii) the training data $\mathcal{E}'$ which consists of positive and negative data and is used in the procedure of section 4.4, and iii) the test data $\mathcal{E}^{ts}$ which consists of positive and negative data and is used for evaluating the selected models in terms of the performance of subcategorization preference test. The sizes of the data sets $\mathcal{E}$, $\mathcal{E}'$, and $\mathcal{E}^{ts}$ are 2,333, 2,100, and 2,100.

5.3 Results

Table 1 shows the performance of subcategorization preference test described in section 4.2.2, for the approximately *optimal* models selected by the procedure in section 4.4 (the "Optimal" model of "General-to-Specific" method), as well as for several other models including baseline models. *Coverage* is the rate of test instances which satisfy the *case covering constraint* of section 4.1. *Accuracy* is measured with the following heuristics: i) verb-noun collocations which satisfy the

[7] *"Agaru (rise)"*, *"kau (buy)"*, *"motoduku (base)"*, *"oujiru (respond)"*, *"sumu (live)"*, *"tigau (differ)"*, and *"tsunagaru (connect)"*.

Table 1: Comparison of Coverage and Accuracy of *Optimal* and Other Models (%)

	Coverage	Accuracy
General-to-Specific		
(Initial)	84.8	81.3
(Independent Cases)	84.8	82.2
(General Classes)	77.5	79.5
(Optimal)	75.4	**87.1**
(MDL)	15.9	70.5
One-by-one Feature Adding		
(Optimal)	60.8	79.0

case covering constraint are preferred, ii) even those verb-noun collocations which do not satisfy the *case covering constraint* are assigned the conditional probabilities in (15) by neglecting cases which are not *covered* by the model. With these heuristics, subcategorization preference can be judged for all the test instances, and test set coverage becomes 100%.

In Table 1, the "Initial" model is the one constructed according to the description in section 4.1, in which cases are independent of each other and the sense restriction of each case is (one of) the most general class(es). The "Independent Cases" model is the one obtained by removing all the case dependencies from the "Optimal" model, while the "General Classes" model is the one obtained by generalizing all the sense restriction of the "Optimal" model to the most general classes. The "MDL" model is the one with the minimum description length. This is for evaluating the effect of the MDL principle in the task of subcategorization preference model learning. The "Optimal" model of "One-by-one Feature Adding" method is the one selected from the sequence of one-by-one feature adding in section 3.1 by the procedure in section 4.4.

The "Optimal" model of 'General-to-Specific" method performs best among all the models in Table 1. Especially, it outperforms the "Optimal" model of "One-by-one Feature Adding" method both in coverage and accuracy. As for the size of the optimal model, the average number of the active feature set is 126 for "General-to-Specific" method and 800 for "One-by-one Feature Adding" method. Therefore, *general-to-specific* feature selection algorithm achieves significant improvements over the *one-by-one feature adding* algorithm with much smaller number of active features. The "Optimal" model of "General-to-Specific" method outperforms both the "Independent Cases" and "General Classes" models, and thus both of the case dependencies and specific sense restriction selected by the proposed method have much contribution to improving the performance in subcategorization preference test. The "MDL" model performs worse than the "Optimal" model, because the features of the "MDL" model have much more specific sense restriction than those of the "Optimal" model, and the coverage of the "MDL" model is much lower than that of the "Optimal" model.

6 Conclusion

This paper proposed a novel method for learning probability models of subcategorization preference of verbs. Especially, we proposed a new model selection algorithm which starts from the most general model and gradually examines more specific models. In the experimental evaluation, it is shown that both of the case dependencies and specific sense restriction selected by the proposed method contribute to improving the performance in subcategorization preference resolution. As for future works, it is important to evaluate the performance of the learned subcategorization preference model in the real parsing task.

References

A. L. Berger, S. A. Della Pietra, and V. J. Della Pietra. 1996. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–71.

E. Charniak. 1997. Statistical Parsing with a Context-free Grammar and Word Statistics. In *Proceedings of the 14th AAAI*, pages 598–603.

M. Collins. 1996. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the 34th Annual Meeting of ACL*, pages 184–191.

S. Della Pietra, V. Della Pietra, and J. Lafferty. 1997. Inducing Features of Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393.

EDR (Japan Electronic Dictionary Research Institute, Ltd.). 1995. *EDR Electronic Dictionary Technical Guide.*

H. Li and N. Abe. 1995. Generalizing Case Frames Using a Thesaurus and the MDL Principle. In *Proceedings of International Conference on Recent Advances in Natural Language Processing*, pages 239–248.

H. Li and N. Abe. 1996. Learning Dependencies between Case Frame Slots. In *Proceedings of the 16th COLING*, pages 10–15.

D. M. Magerman. 1995. Statistical Decision-Tree Models for Parsing. In *Proceedings of the 33rd Annual Meeting of ACL*, pages 276–283.

NLRI (National Language Research Institute). 1993. *Word List by Semantic Principles.* Syuei Syuppan. (in Japanese).

P. Resnik. 1993. Semantic Classes and Syntactic Ambiguity. In *Proceedings of the Human Language Technology Workshop*, pages 278–283.

J. Rissanen. 1984. Universal Coding, Information, Prediction, and Estimation. *IEEE Transactions on Information Theory*, IT-30(4):629–636.

T. Utsuro and Y. Matsumoto. 1997. Learning Probabilistic Subcategorization Preference by Identifying Case Dependencies and Optimal Noun Class Generalization Level. In *Proceedings of the 5th ANLP*, pages 364–371.