A Statistical Learning Approach to Lexical Knowledge Acquisition and Structural Disambiguation

Hang Li
Microsoft Research China
5F, Beijing Sigma Center No.49, Zhichun Road, Haidian District, Beijing 100080
Email: hangli@microsoft.com

Abstract: In this paper, we consider the problem of generalizing case frame slots (or learning case slot patterns). We formalize this problem as that of estimating a probability model, which we call case slot model. We restrict the class of case slot models to that of tree cut models by using an existing thesaurus. In this way, the problem of generalizing the values of a case slot turns out to be that of estimating a model from the class of tree cut models for some fixed thesaurus tree. We employ the Minimum Description Length (MDL) principle for model estimation. We then employ an efficient algorithm, which provably obtains the optimal tree cut model in terms of MDL.
Keyword: Statistical Language Processing, MDL Principle, Case Frame, Thesaurus.

用于词汇知识获取和结构消歧的一种统计学习方法

李 航
微软中国研究院
北京海淀区知春路 49 号北京希格玛中心 5 楼，100080

摘要：在本文中我们考虑格框架槽的一般化（或学习）问题。我们把这个问题看作是对一个概率模型的估值问题，并称此模型为格槽模型。我们用一部现有的义类树把格槽模型的类型限制为树分割模型的类型。在这种方式下，一个格槽值的一般化问题就转化为在一棵固定的义类树上的树分割模型的模型估值的问题。我们采用最小描述长度（MDL）原理来进行模型估值。估值时应用了一种高效率的算法，并可以证明用这种方法获得的模型在 MDL 意义上是最佳的。
关键词： 概率语言处理，MDL 原理，格框架，义类树

1. Introduction

Structural disambiguation in sentence analysis is still a central problem in natural language processing. Past researches have verified that using lexical semantic knowledge can, to a quite large extent, cope with this problem. Although there have been many studies conducted in the past to address the lexical knowledge acquisition problem, further investigation, especially that based on a *principled methodology* is still needed, and this is, in fact, the problem we address in this paper.

The problem of acquiring and using lexical semantic knowledge, especially that of case frame patterns, can be formalized as follows. A learning module acquires case frame patterns on the basis of some case frame instances extracted from corpus data. A processing (disambiguation) module then refers to the acquired knowledge and judges the degrees of acceptability of some number of new case frames, including previously unseen ones.

In this paper, we consider the problem of generalizing the values of a case frame slot (or learning case slot patterns). We formalize this problem as that of estimating a probability model from a class of models, which we call case slot models. We next restrict the class of case slot models to that of what we refer to as tree cut models by using an existing thesaurus. In this way, the problem of generalizing the values of a case slot turns out to be that of estimating a model from the class of tree cut models for some fixed thesaurus tree. We employ the Minimum Description Length (MDL) principle for model estimation, and employ an efficient algorithm, which provably obtains the optimal tree cut model in terms of MDL.

2. Case Slot Model

Table 1. Example Case Slot Data

Verb	Slot Name	Slot Value	Frequency
fly	arg1	crow	2
fly	arg1	eagle	2
fly	arg1	bird	4
fly	arg1	bee	2

We can assume that case slot data for a case slot for a verb like that shown in Table 1 are generated according to a conditional probability distribution, which specifies the conditional probability of a noun given the verb and the case slot. We call such a distribution a `case slot model.' More precisely, the conditional probability of a noun is defined as that of the noun class to which the noun belongs, divided by the size of the noun class.

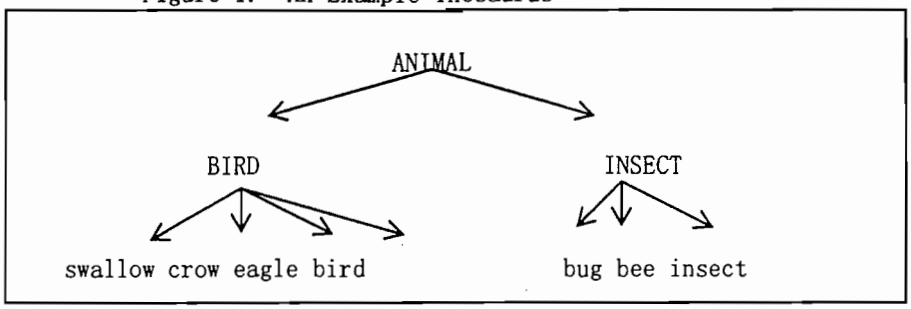
Suppose that N is the set of nouns, V is the set of verbs, and R is the set of slot names. A partition Π of N is defined as a set satisfying $\Pi \subseteq 2^N, \bigcup_{C \in \Pi} C = N, \forall C_i, C_j \in \Pi, C_i \bigcap C_j = \phi, (i \neq j)$. A case slot model with respect to a partition Π is defined as a conditional probability distribution:

$$P(n \mid v, r) = \frac{1}{|C|} P(C \mid v, r), n \in C$$

where random variable n assumes a value from N, random variable v from V, and random variable r from R, and where $C \in \Gamma$ is satisfied.

In this way, the problem of generalizing case frame slots can be formalized as that of selecting a model from a class of case slot models. Since the number of partitions for a set of nouns is of exponential order, the problem of estimating a case slot model is most likely intractable.

Figure 1. An Example Thesaurus

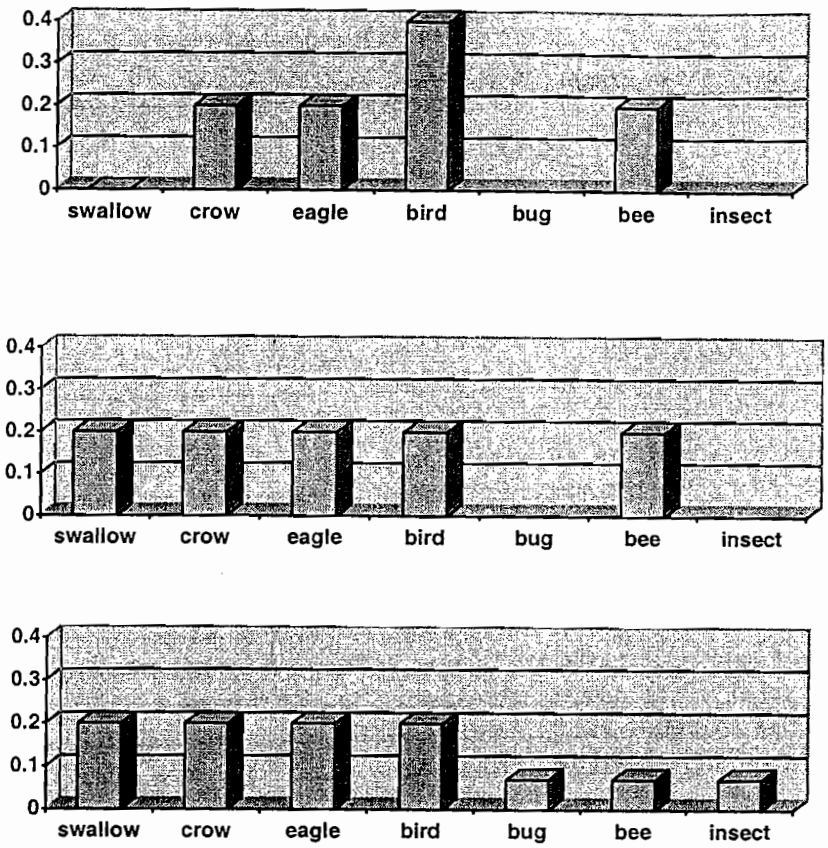


To deal with this difficulty, we take the approach of restricting the class of case slot models. We reduce the number of partitions necessary for consideration by using a thesaurus. Specifically, we restrict attention to those partitions that exist within the thesaurus in the form of a cut. Here by thesaurus is meant a rooted tree in which each leaf node stands for a noun, while each internal node represents a noun class, and a directed link represents set inclusion (cf., Figure 1). A cut in a tree is any set of nodes in the tree that can represent a partition of the given set of nouns. For example, in the thesaurus of Figure 1, there are five cuts: [ANIMAL], [BIRD, INSECT], [BIRD, bug, bee, insect], [swallow, crow, eagle, bird, INSECT], [swallow, crow, eagle, bird, bug, bee, insect].

If we employ MLE for parameter estimation, we can obtain five tree cut models from the case slot data in Table 1; Figure 2 shows three of these. For example, the second model in Figure 2 is one such tree cut model. Recall that the tree cut model M defines a conditional probability distribution $P_M(n|v,r)$ in the following way: for any noun that is in the tree cut, such as `bee,' the probability is given as explicitly specified by the model, i.e., $P_M(\text{bee}|\text{fly},\text{arg1}) = 0.2$; for any class in the tree cut, the probability is distributed uniformly to all nouns included in it. For example, since there are four nouns that fall under the class BIRD, and `swallow' is one of them, the probability of `swallow' is thus given by $P_M(\text{swallow}|\text{fly},\text{arg1}) = 0.8/4 = 0.2$. Note that the probabilities assigned to the nouns under BIRD are *smoothed*, even if the nouns have different observed frequencies.

In this way, the problem of generalizing the values of a case slot has been formalized into that of estimating a model from the class of tree cut models for some fixed thesaurus tree.

Figure 2. Tree Cut Models



3. MDL as Strategy

The question now becomes what strategy (criterion) we should employ to select the best tree cut model. We propose to adopt the MDL principle.

The Minimum Description Length (MDL) principle is a strategy (criterion) for data compression and statistical estimation. MDL states that for statistical estimation, the best probability model with respect to given data is that which requires the shortest code length in bits for encoding the model itself and the data observed through it. (For an introduction to MDL, see [4]).

In our current problem, a model nearer the root of the thesaurus tree, such as the third model in Figure 2, generally tends to be simpler (in terms of the number of parameters), but also tends to have a poorer fit to the data. By way of contrast, a model nearer the leaves of the thesaurus tree, such as the first model, tends to be more complex, but also tends to have a better fit to the data. Thus, there is a trade-off between the simplicity of a model and the goodness of its fit to the data. The use of MDL can balance the trade-off relationship.

Let us consider how to calculate description length for the current problem. Given a sample S and a tree cut Γ, we can employ MLE to estimate the parameters of the corresponding tree cut model M.

The total description length L of the tree cut model M and the data S observed through it may be computed as the sum of model description length Lm, and data description length Ld, i.e., $L = Lm + Ld$.

Model description length Lm may be calculated by

$$Lm = \frac{k}{2} \log |S|$$

where $|S|$ denotes the sample size and k denotes the number of free parameters in the tree cut model, i.e., k equals the number of nodes in Γ minus one.

Finally, data description length Ld may be calculated as

$$Ld = -\sum_{n \in S} \log P(n)$$

where for simplicity we write $P(n)$ for $P_M(n|v,r)$. Recall that $P(n)$ is obtained by MLE, i.e.,

$$P(n) = \frac{1}{|C|} P(C)$$

for each $n \in C$, where for each $C \in \Gamma$

$$P(C) = \frac{f(C)}{|S|}$$

where $f(C)$ denotes the frequency of nouns in class C in data S.

With the description length defined in the above manner, we wish to select a model with the minimum description length, and then output it as the result of generalization.

4. Algorithm

In generalizing the values of a case slot using MDL, if computation time were of no concern, one could in principle calculate the description length for every possible tree cut model and output a model with the minimum description length as a generalization result, but since the number of cuts in a thesaurus tree is usually exponential, it is impractical to do so. Nonetheless, we were able to devise a simple and efficient algorithm, based on dynamic programming, which is guaranteed to find a model with the minimum description length.

The algorithm, which we call Find-MDL, recursively finds the optimal submodel for each child subtree of a given (sub)tree and follows one of two possible courses of action: (1) it either combines these optimal submodels and returns this

combination as output, or (2) it collapses all these optimal submodels into the (sub)model containing the root node of the given (sub)tree. Find-MDL simply chooses the course of action which will result in the shorter description length (cf., Figure 3). Note that for simplicity we describe Find-MDL as outputting a tree cut, rather than a tree cut model.

Figure 3. Algorithm: Find-MDL

Let t denote a thesaurus (sub)tree, while root (t) denotes the root of t. Let c denote a tree cut in t. Initially t is set to the entire tree. Find-MDL(t):= c

```
1.   if (t is a leaf node)
2.   then
3.      return([t]);
4.   else
5.      For each child subtree t_i of t c_i :=Find-MDL(t_i);
6.      c:= append(c_i);
7.      if (L'([root (t)]) < L'(c))
8.      then
9.         return([root (t)]);
10.     else
11.        return(c).
```

Concerning the above algorithm, the following proposition holds:

Proposition 1 The algorithm Find-MDL terminates in time O(N), where N denotes the number of leaf nodes in the thesaurus tree T, and it outputs a tree cut model of T with the minimum description length.

Using the MDL-based method described above, we can generalize the values of a case slot. The probability of a noun being the value of a slot can then be represented as a conditional probability estimated (smoothed) from a class-based model on the basis of the MDL principle.

The case slot generalization problem can also be considered as that of generalizing individual nouns present in case slot data into classes of nouns present in a given thesaurus. For example, given the thesaurus in Figure 1 and frequency data in Table 1, we would like our system to judge that the class `BIRD' and the noun `bee' can be the value of the arg1 slot for the verb `fly.' The problem of deciding whether to stop generalizing at `BIRD' and `bee' or to continue generalizing further to `ANIMAL' has been addressed by a number of researchers. The MDL-based method described above provides a disciplined way to realize this on the basis of data compression and statistical estimation.

The MDL-based method, in fact, conducts generalization in the following way. When the differences between the frequencies of the words in a class are not

large enough (relative to the entire data size and the number of the words), it generalizes them into the class. When the differences are especially noticeable (relative to the entire data size and the number of the words), on the other hand, it stops generalization at that level.

5. Summary

We have proposed a method for generalizing case slots. The method has the following merits: (1) it is theoretically sound; (2) it is computationally efficient; (3) it is robust against noise. Experimental results indicate that our method is a very effective in case slot generalization and structural disambiguation (for further detail, see [1, 4]). We have also developed several other learning methods for lexical knowledge acquisition (cf., [2, 3, 4]). All of the experimental results indicate that the use of MDL is a very promising approach in natural language processing.

Acknowledgements

I would like to express my sincere gratitude to Prof. C.N. Huang of Microsoft for his much helpful advices with regard to the writing of this paper.

Reference

[1]. Hang Li and Naoki Abe, Generalizing Case Frames Using a Thesaurus and the MDL Principle, Computational Linguistics 24(2), 217-244 (1998).

[2]. Hang Li and Naoki Abe, Learning Dependencies between Case Frame Slots, Computational Linguistics 25(2), 283-291 (1999)

[3]. Hang Li and Naoki Abe, Word Clustering and Disambiguation Based on Co-occurrence Data, Proc. of COLING-ACL' 98, 749-755.

[4]. Hang Li, A Probabilistic Approach to Lexical Semantic Knowledge Acquisition and Structural Disambiguation. PhD thesis, the Univ. of Tokyo. (1998).