Improving the Accuracy of Subcategorizations Acquired from Corpora

Naoki Yoshinaga
Department of Computer Science,
University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-0033
yoshinag@is.s.u-tokyo.ac.jp

Abstract

This paper presents a method of improving the accuracy of subcategorization frames (SCFs) acquired from corpora to augment existing lexicon resources. I estimate a confidence value of each SCF using corpus-based statistics, and then perform clustering of SCF confidence-value vectors for words to capture co-occurrence tendency among SCFs in the lexicon. I apply my method to SCFs acquired from corpora using lexicons of two large-scale lexicalized grammars. The resulting SCFs achieve higher precision and recall compared to SCFs obtained by naive frequency cut-off.

1 Introduction

Recently, a variety of methods have been proposed for acquisition of subcategorization frames (SCFs) from corpora (surveyed in (Korhonen, 2002)). One interesting possibility is to use these techniques to improve the coverage of existing large-scale lexicon resources such as lexicons of lexicalized grammars. However, there has been little work on evaluating the impact of acquired SCFs with the exception of (Carroll and Fang, 2004).

The problem when we integrate acquired SCFs into existing lexicalized grammars is lower quality of the acquired SCFs, since they are acquired in an unsupervised manner, rather than being manually coded. If we attempt to compensate for the poor precision by being less strict in filtering out less likely SCFs, then we will end up with a larger number of *noisy* lexical entries, which is problematic for parsing with lexicalized grammars (Sarkar et al., 2000). We thus need some method of selecting the most reliable set of SCFs from the system output as demonstrated in (Korhonen, 2002).

In this paper, I present a method of improving the accuracy of SCFs acquired from corpora in order to augment existing lexicon resources. I first estimate a confidence value that a word can have each SCF, using corpus-based statistics. To capture latent co-occurrence tendency among SCFs in the target lexicon, I next perform clustering of SCF confidence-value vectors of words in the acquired lexicon and the target lexicon. Since each centroid value of the obtained clusters indicates whether the words in that cluster have each SCF, we can eliminate SCFs acquired in error and predict possible SCFs according to the centroids.

I applied my method to SCFs acquired from a corpus of newsgroup posting about mobile phones (Carroll and Fang, 2004), using the XTAG English grammar (XTAG Research Group, 2001) and the LinGO English Resource Grammar (ERG) (Copestake, 2002). I then compared the resulting SCFs with SCFs obtained by naive frequency cut-off to observe the effects of clustering.

2 Background

2.1 SCF Acquisition for Lexicalized Grammars

I start by acquiring SCFs for a lexicalized grammar from corpora by the method described in (Carroll and Fang, 2004).

```
#S(EPATTERN :TARGET |yield|
            :SUBCAT (VSUBCAT NP)
            :CLASSES ((24 51 161) 5293)
            :RELIABILITY 0
            :FREQSCORE 0.26861903
            :FREQCNT 1 :TLTL (VV0)
            :SLTL ((|route| NN1))
            :OLT1L ((|result| NN2))
            :OLT2L NIL
            :OLT3L NIL :LRL 0))
```

Figure 1: An acquired SCF for a verb "*yield*"

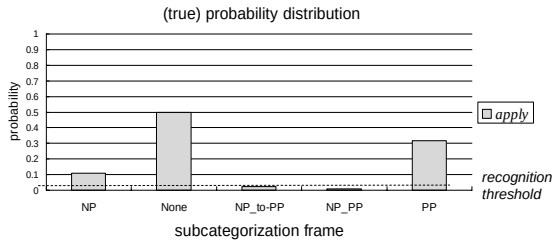


Figure 2: SCF probability distributions for *apply*

In their study, they first acquire fine-grained SCFs using the unsupervised method proposed by Briscoe and Carroll (1997) and Korhonen (2002). Figure 1 shows an example of one acquired SCF entry for a verb "*yield*." Each SCF entry has several fields about the observed SCF. I explain here only its portion related to this study. The TARGET field is a word stem, the first number in the CLASSES field indicates an SCF type, and the FREQCNT field shows how often words derivable from the word stem appeared with the SCF type in the training corpus. The obtained SCFs comprise the total 163 SCF types which are originally based on the SCFs in the ANLT (Boguraev and Briscoe, 1987) and COMLEX (Grishman et al., 1994) dictionaries. In this example, the SCF type 24 corresponds to an SCF of transitive verb.

They then obtain SCFs for the target lexicalized grammar (the LinGO ERG (Copestake, 2002) in their study) using a handcrafted translation map from these 163 types to the SCF types in the target grammar. They reported that they could achieve a coverage improvement of 4.5% but that average parse time was doubled. This is because they did not use any filtering method for the acquired SCFs to suppress an increase of the lexical ambiguity. We definitely need some method to control the quality of the acquired SCFs.

Their method is extendable to any lexicalized grammars, if we could have a translation map from these 163 types to the SCF types in the grammar.

2.2 Clustering of Verb SCF Distributions

There is some related work on clustering of verbs according to their SCF probability distributions (Schulte im Walde and Brew, 2002; Korhonen et al., 2003). Schulte im Walde and Brew (2002) used the k-Means (Forgy, 1965) algorithm to cluster SCF distributions for monosemous verbs while Korhonen et al. (2003) applied other clustering methods to cluster polysemic SCF data. These studies aim at obtaining verb semantic classes, which are closely related to syntactic behavior of argument selection (Levin, 1993).

Korhonen (2002) made use of SCF distributions for representative verbs in Levin's verb classes to obtain accurate back-off estimates for all the verbs in the classes. In this study, I assume that there are classes whose element words have identical SCF types. I then obtain these classes by clustering acquired SCFs, using information available in the target lexicon, and directly use the obtained classes to eliminate implausible SCFs.

3 Method

3.1 Estimation of Confidence Values for SCFs

I first create *an SCF confidence-value vector* v_i for each word w_i, an object for clustering. Each element v_{ij} in v_i represents a confidence value of SCF s_j for a word w_i, which expresses how strong the evidence is that the word w_i has SCF s_j. Note that a confidence value $conf_{ij}$ is not a probability that a word w_i appears with SCF s_j but a probability of existence of SCF s_j for the word w_i. In this study, I assume that a word w_i appears with each SCF s_j with a certain (non-zero) probability $\theta_{ij} (= p(s_{ij}|w_i) > 0$ where $\sum_j \theta_{ij} = 1)$, but only SCFs whose probabilities exceed a certain threshold are recognized in the lexicon. I hereafter call this threshold *recognition threshold*. Figure 2 depicts a probability distribution of SCF for *apply*. In this context, I can regard a confidence value of each SCF as a probability that the probability of that SCF exceeds the recognition threshold.

One intuitive way to estimate a confidence value is to assume an observed probability, *i.e.*, relative frequency, is equal to a probability θ_{ij} of SCF s_j for a word w_i ($\theta_{ij} = freq_{ij} / \sum_j freq_{ij}$ where $freq_{ij}$ is a frequency that a word w_i appears with SCF s_j in corpora). When the relative frequency of s_j for a word w_i exceeds the recognition threshold, its confidence value $conf_{ij}$ is set to 1, and otherwise $conf_{ij}$ is set to 0. However, an observed probability is unreliable for infrequent words. Moreover, when we want to encode confidence values of reliable SCFs in the target grammar, we cannot distinguish the confidence values of those SCFs with confidence values of acquired SCFs.

The other promising way to estimate a confidence value, which I adopt in this study, is to assume a probability θ_{ij} as a stochastic variable in the context of Bayesian statistics (Gelman et al., 1995). In this context, *a posteriori* distribution of the probability θ_{ij} of an SCF s_j for a word w_i is given by:

$$p(\theta_{ij}|D) = \frac{P(\theta_{ij})P(D|\theta_{ij})}{P(D)} = \frac{P(\theta_{ij})P(D|\theta_{ij})}{\int_0^1 P(\theta_{ij})P(D|\theta_{ij})d\theta_{ij}}, \quad (1)$$

where $P(\theta_{ij})$ is *a priori* distribution, and D is the data we have observed. Since every occurrence of SCFs in the data D is independent with each other, the data D can be regarded as Bernoulli trials. When we observe the data D that a word w_i appears n times in total and $x (\leq n)$ times with SCF s_j,[1] its conditional distribution is represented by binominal distribution:

$$P(D|\theta_{ij}) = \binom{n}{x} \theta_{ij}^x (1-\theta_{ij})^{(n-x)}. \quad (2)$$

To calculate this *a posteriori* distribution, I need to define the *a priori* distribution $P(\theta_{ij})$. The question is which probability distribution of θ_{ij} can appropriately reflects prior knowledge. In other words, it should encode knowledge we use to estimate SCFs for unknown words. I simply determine it from distributions of observed probability values of s_j for words seen in corpora[2] by using a method described in (Tsuruoka and Chikayama, 2001). In their study, they assume *a priori* distribution as the *beta* distribution defined as:

$$p(\theta_{ij}|\alpha,\beta) = \frac{\theta_{ij}^{\alpha-1}(1-\theta_{ij})^{\beta-1}}{B(\alpha,\beta)}, \quad (3)$$

where $B(\alpha,\beta) = \int_0^1 \theta_{ij}^{\alpha-1}(1-\theta_{ij})^{\beta-1}d\theta_{ij}$. The value of α and β is determined by moment estimation.[3] By substituting Equations 2 and 3 into Equation 1, I finally obtain the *a posteriori* distribution $p(\theta_{ij}|D)$ as:

$$p(\theta_{ij}|\alpha,\beta,D) = c \cdot \theta_{ij}^{x+\alpha-1}(1-\theta_{ij})^{n-x+\beta-1}, \quad (4)$$

where $c = \binom{n}{x} / (B(\alpha,\beta) \int_0^1 P(\theta_{ij})P(D|\theta_{ij})d\theta_{ij})$.

When I regard the recognition threshold as t, I can calculate a confidence value $conf_{ij}$ that a word w_i can have s_j by integrating the *a posteriori* distribution $p(\theta_{ij}|D)$ from the threshold t to 1:

$$conf_{ij} = \int_t^1 c \cdot \theta_{ij}^{x+\alpha-1}(1-\theta_{ij})^{n-x+\beta-1}d\theta_{ij}. \quad (5)$$

By using this confidence value, I represent an SCF confidence-value vector v_i for a word w_i in the acquired SCF lexicon ($v_{ij} = conf_{ij}$).

In order to combine SCF confidence-value vectors for words acquired from corpora and those for words in the lexicon of the target grammar, I also represent an SCF confidence-value vector v'_i for a word w'_i in the target grammar by:

$$v'_{ij} = \begin{cases} 1-\varepsilon & w'_i \text{ has } s_j \text{ in the lexicon} \\ \varepsilon & \text{otherwise}, \end{cases} \quad (6)$$

where ε expresses an unreliability of the lexicon. In this study, I trust the lexicon as much as possible by setting ε to the machine epsilon.

3.2 Clustering of SCF Confidence-Value Vectors

I next present a clustering algorithm of words according to their SCF confidence-value vectors. Given k initial representative vectors called *centroids*, my algorithm iteratively updates clusters by assigning each data object to its closest centroid

[1] The values of `FREQCNT` is used to obtain n and x.

[2] I estimated *a priori* distribution separately for each type of SCF from words that appeared more than 50 times in the training corpus in the following experiments.

[3] The expectation and variance of the *beta* distribution are made equal to those of the observed probability values.

```
Input:  a set of SCF confidence-value
          vectors V = {v_1, v_2, ..., v_n} ⊆ R^m
        a distance function d : R^m × Z^m → R
        a function to compute a centroid
          μ : {v_j1, v_j2, ..., v_jl} → Z^m
        initial centroids C = {c_1, c_2, ..., c_k} ⊆ Z^m
Output: a set of clusters {C_j}

while cluster members are not stable do
  foreach cluster C_j
    C_j = {v_i | ∀c_l, d(v_i, c_j) ≥ d(v_i, c_l)}          (1)
  end foreach
  foreach clusters C_j
    c_j = μ(C_j)                                          (2)
  end foreach
end while

return {C_j}
```

Figure 3: Clustering algorithm for SCF confidence-value vectors

and recomputing centroids until cluster members become stable, as depicted in Figure 3.

Although this algorithm is roughly based on the k-Means algorithm, it is different from k-Means in important respects. I assume the elements of the centroids of the clusters as a discrete value of 0 or 1 because I want to obtain clusters whose element words have the exactly same set of SCFs.

I then derive a distance function d to calculate a probability that a data object v_i should have an SCF set represented by a centroid c_m as follows:

$$d(v_i, c_m) = \prod_{c_{mj}=1} v_{ij} \cdot \prod_{c_{mj}=0} (1 - v_{ij}). \quad (7)$$

By using this function, I can determine the closest cluster as $\underset{C_m}{\operatorname{argmax}}\, d(v_i, c_m)$ (`(1)` in Figure 3).

After every assignment, I calculate a next centroid c_m of each cluster C_m (`(2)` in Figure 3) by comparing a probability that the words in the cluster have an SCF s_j and a probability that the words in the cluster do not have the SCF s_j as follows:

$$c_{mj} = \begin{cases} 1 & \textit{when} \prod_{v_i \in C_m} v_{ij} > \prod_{v_i \in C_m} (1 - v_{ij}) \\ 0 & \textit{otherwise}. \end{cases} \quad (8)$$

I next address the way to determine the number of clusters and initial centroids. In this study, I assume that the most of the possible set of SCFs for words are included in the lexicon of the target grammar,[4] and make use of the existing sets of SCFs for the words in the lexicon to determine the number of clusters and initial centroids. I first extract SCF confidence-value vectors from the lexicon of the grammar. By eliminating duplications from them and regarding $\varepsilon = 0$ in Equation 6, I obtain initial centroids c_m. I then initialize the number of clusters k to the number of c_m.

I finally update the acquired SCFs using the obtained clusters and the confidence values of SCFs in this order. I call the following procedure *centroid cut-off t* when the confidence values are estimated under the recognition threshold t. Since the value c_{mj} of a centroid c_m in a cluster C_m represents whether the words in the cluster can have SCF s_j, I first obtain SCFs by collecting SCF s_j for a word $w_i \in C_m$ when c_{mj} is 1. I then eliminate implausible SCFs s_j for w_i from the resulting SCFs according to their confidence values $conf_{ij}$.

In the following, I compare centroid cut-off with *frequency cut-off* and *confidence cut-off t*, which use relative frequencies and confidence values calculated under the recognition threshold t, respectively. Note that these cut-offs use only corpus-based statistics to eliminate SCFs.

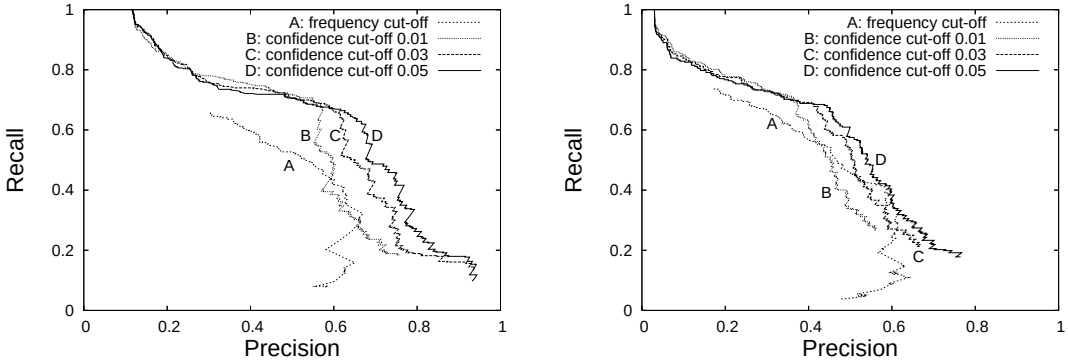
4 Experiments

I applied my method to SCFs acquired from 135,902 sentences of mobile phone newsgroup postings archived by Google.com, which is the same data used in (Carroll and Fang, 2004). The number of acquired SCFs was 14,783 for 3,864 word stems, while the number of SCF types in the data was 97. I then translated the 163 SCF types into the SCF types of the XTAG English grammar (XTAG Research Group, 2001) and the LinGO ERG (Copestake, 2002)[5] using translation mappings built by Ted Briscoe and Dan Flickinger from 23 of the SCF types into 13 (out of 57 possible) XTAG SCF types, and 129 into 54 (out of 216 possible) ERG SCF types.

To evaluate my method, I split each lexicon of the two grammars into the training SCFs and the testing SCFs. The words in the testing SCFs were included in the acquired SCFs. When I apply my method to the acquired SCFs using the training SCFs and evaluate the resulting SCFs with the

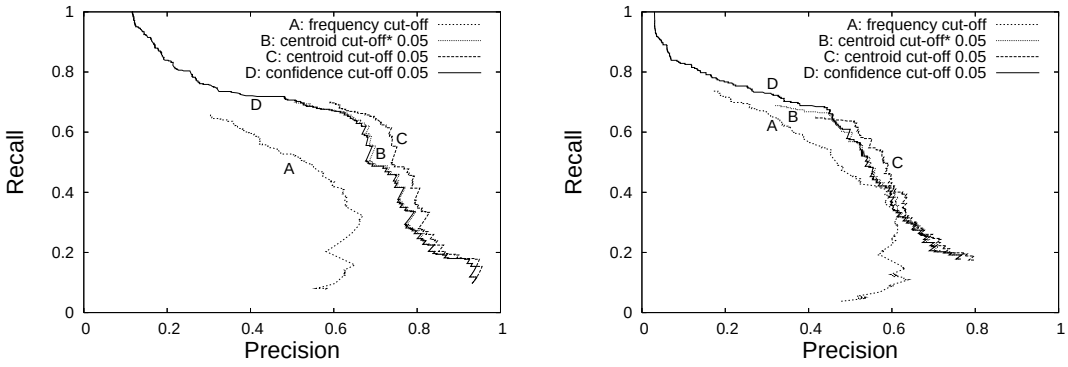
[4] When the lexicon is less accurate, I can determine the number of clusters using other algorithms (Hamerly, 2003).

[5] I used the same version of the LinGO ERG as (Carroll and Fang, 2004) (1.4; April 2003) but the map is updated.



XTAG ERG

Figure 4: Precision and recall of the resulting SCFs using confidence cut-offs and frequency cut-off: the XTAG English grammar (left) the LinGO ERG (right)



XTAG ERG

Figure 5: Precision and recall of the resulting SCFs using confidence cut-off and centroid cut-off: the XTAG English grammar (left) the LinGO ERG (right)

testing SCFs, we can estimate to what extent my method can preserve reliable SCFs for words unknown to the grammar.[6] The XTAG lexicon was split into 9,437 SCFs for 8,399 word stems as training and 423 SCFs for 280 word stems as testing, while the ERG lexicon was split into 1,608 SCFs for 1,062 word stems as training and 292 SCFs for 179 word stems as testing. I extracted SCF confidence-value vectors from the training SCFs and the acquired SCFs for the words in the testing SCFs. The number of the resulting data objects was 8,679 for XTAG and 1,241 for ERG.

The number of initial centroids[7] extracted from the training SCFs was 49 for XTAG and 53 for ERG. I then performed clustering of 8,679 data objects into 49 clusters and 1,241 data objects into 53 clusters, and then evaluated the resulting SCFs by comparing them to the testing SCFs.

I first compare confidence cut-off with frequency cut-off to observe the effects of Bayesian estimation. Figure 4 shows precision and recall of the SCFs obtained using frequency cut-off and confidence cut-off 0.01, 0.03, and 0.05 by varying threshold for the confidence values and the relative frequencies from 0 to 1.[8] The graph indicates that the confidence cut-offs achieved higher recall than the frequency cut-off, thanks to the *a priori* distributions. When we compare the three confidence cut-offs, we can improve precision using higher recognition thresholds while we can improve recall using lower recognition thresholds. This is quite consistent with our expectations.

[6] I here assume that the existing SCFs for the words in the lexicon are more reliable than the other SCFs for those words.

[7] I used the vectors that appeared for more than one word.

[8] $\text{Precision} = \frac{\text{Correct SCFs for the words in the resulting SCFs}}{\text{All SCFs for the words in the resulting SCFs}}$

$\text{Recall} = \frac{\text{Correct SCFs for the words in the resulting SCFs}}{\text{All SCFs for the words in the test SCFs}}$

I then compare centroid cut-off with confidence cut-off to observe the effects of clustering. Figure 5 shows precision and recall of the resulting SCFs using centroid cut-off 0.05 and the confidence cut-off 0.05 by varying the threshold for the confidence values. In order to show the effects of the use of the training SCFs, I also performed clustering of SCF confidence-value vectors in the acquired SCFs with random initialization (k = 49 (for XTAG) and 53 (for ERG); centroid cut-off 0.05*). The graph shows that clustering is meaningful only when we make use of the reliable SCFs in the manually-coded lexicon. The centroid cut-off using the lexicon of the grammar boosted precision compared to the confidence cut-off.

The difference between the effects of my method on XTAG and ERG would be due to the finer-grained SCF types of ERG. This resulted in lower precision of the acquired SCFs for ERG, which prevented us from distinguishing infrequent (correct) SCFs from SCFs acquired in error. However, since unusual SCFs tend to be included in the lexicon, we will be able to have accurate clusters for unknown words with smaller SCF variations as we achieved in the experiments with XTAG.

5 Concluding Remarks and Future Work

In this paper, I presented a method to improve the quality of SCFs acquired from corpora using existing lexicon resources. I applied my method to SCFs acquired from corpora using lexicons of the XTAG English grammar and the LinGO ERG, and have shown that it can eliminate implausible SCFs, preserving more reliable SCFs.

In the future, I need to evaluate the quality of the resulting SCFs by manual analysis and by using the extended lexicons to improve parsing. I will investigate other clustering methods such as hierarchical clustering, and use other information for clustering such as semantic preference of arguments of SCFs to have more accurate clusters.

Acknowledgments

I thank Yoshimasa Tsuruoka and Takuya Matsuzaki for their advice on probabilistic modeling, Alex Fang for his help in using the acquired SCFs, and Anna Korhonen for her insightful suggestions on evaluation. I am also grateful to Jun'ichi Tsujii, Yusuke Miyao, John Carroll and the anonymous reviewers for their valuable comments. This work was supported in part by JSPS Research Fellowships for Young Scientists and in part by CREST, JST (Japan Science and Technology Agency).

References

B. Boguraev and T. Briscoe. 1987. Large lexicons for natural language processing: utilising the grammar coding system of LDOCE. *Computational Linguistics*, 13(4):203–218.

T. Briscoe and J. Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proc. the fifth ANLP*, pages 356–363.

J. Carroll and A. C. Fang. 2004. The automatic acquisition of verb subcategorizations and their impact on the performance of an HPSG parser. In *Proc. the first ijc-NLP*, pages 107–114.

A. Copestake. 2002. *Implementing typed feature structure grammars*. CSLI publications.

E. W. Forgy. 1965. Cluster analysis of multivariate data: Efficiency vs. interpretability of classifications. *Biometrics*, 21:768–780.

A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, editors. 1995. *Bayesian Data Analysis*. Chapman and Hall.

R. Grishman, C. Macleod, and A. Meyers. 1994. Comlex syntax: Building a computational lexicon. In *Proc. the 15th COLING*, pages 268–272.

G. Hamerly. 2003. *Learning structure and concepts in data through data clustering*. Ph.D. thesis, University of California, San Diego.

A. Korhonen, Y. Krymolowski, and Z. Marx. 2003. Clustering polysemic subcategorization frame distributions semantically. In *Proc. the 41st ACL*, pages 64–71.

A. Korhonen. 2002. *Subcategorization Acquisition*. Ph.D. thesis, University of Cambridge.

B. Levin. 1993. *English Verb Classes and Alternations*. Chicago University Press.

A. Sarkar, F. Xia, and A. K. Joshi. 2000. Some experiments on indicators of parsing complexity for lexicalized grammars. In *Proc. the 18th COLING workshop*, pages 37–42.

S. Schulte im Walde and C. Brew. 2002. Inducing German semantic verb classes from purely syntactic subcategorisation information. In *Proc. the 41st ACL*, pages 223–230.

Y. Tsuruoka and T. Chikayama. 2001. Estimating reliability of contextual evidences in decision-list classifiers under Bayesian learning. In *Proc. the sixth NLPRS*, pages 701–707.

XTAG Research Group. 2001. A Lexicalized Tree Adjoining Grammar for English. Technical Report IRCS-01-03, IRCS, University of Pennsylvania.