Selectional Preference Based Verb Sense Disambiguation Using WordNet

Patrick Ye

Department of Computer Science and Software Engineering
University of Melbourne, VIC 3010, Australia
jingy@cs.mu.oz.au

Abstract

Selectional preferences are a source of linguistic information commonly applied to the task of Word Sense Disambiguation (WSD). To date, WSD systems using selectional preferences as the main disambiguation mechanism have achieved limited success. One possible reason for this limitation is the limited number of semantic roles used in the construction of selectional preferences. This study investigates whether better performance can be achieved using the current state-of-art semantic role labelling systems, and explores alternative ways of applying selectional preferences for WSD. In this study, WordNet noun synonym sets and hypernym sets were used in the construction of selectional preferences; Semcor2.0 data was used for the training and evaluation of a support vector machine classifier and a Naive Bayes classifier.

1 Introduction

Word Sense Disambiguation (WSD) is the process of examining word tokens in a given context and specifying exactly which sense of each word is intended in that context. It has many applications in natural language processing related areas such as document retrieval, question answering, and compositional sentence analysis (Jurafsky and Martin, 2000), to name a few.

WSD systems can be roughly divided into two categories based on how the disambiguation information is obtained and applied: knowledge based and corpus based. Knowledge based systems in general require certain existing linguistic information repositories which provides all the information that can be used by the disambiguation system to distinguish different senses of the same polysemous words based on the context. Examples of knowledge based systems include dictionary based systems (Lesk, 1986) and selectional preference based systems (Resnik, 1997).

Corpus based systems in general do not require any linguistic information, instead, they require a certain amount of training data (labelled or unlabelled), and a set of predefined disambiguation features which can be used by a statistical method to train a classifier which then is used in the disambiguation of previously unseen data. A corpus based system is described in (Yarowsky, 1995).

Selectional preferences between predicating words (verbs and adjectives) and their arguments (nouns) are a type of linguistic information which has previously been combined with statistical methods to perform word sense disambiguation, ((Resnik, 1997) and (McCarthy and Carroll, 2003)). A selectional preference is a function mapping semantic-role to noun type. The basic assumption made by all selectional preference based WSD systems is that the different senses of the same predicating word would have different selectional preferences with their arguments.

As will be discussed in section 2.2, selectional preference based WSD systems developed so far are limited in terms of coverage and accuracy. In my opinion, the most important cause of this limitation is these systems' inability to extract a sufficient number of semantic roles to be used in the construction of selectional preferences. For example, if a WSD system uses only the subject of the verbs in the selectional preferences, then it cannot be expected to correctly identify the appropriate sense of the verb "run" in "John ran a race" and "John ran a restaurant", since the distinguishing feature of these two senses of "run" comes from the objects they take.

Given the difficulty of semantic role labelling, it is not surprising that only a small set of semantic roles have been used in the literature on selectional preference based WSD. However, recent developments in semantic role labelling makes it possible to extract a much richer set of semantic roles from unrestricted text, thereby enabling more complex selectional preferences to be constructed.

The main objective of this study is to investigate whether the performance of selectional preference based WSD can be improved by using the current state-of-art semantic role labelling systems. This

paper is organised as follows: section 2 will give a formal description of the research problem presented in this paper; section 3 will provide a review of some related work; section 4 will discuss the statistical methods investigated in this study and how they are combined with selectional preferences; the results of this study will be presented in section 5; and section 6 gives a conclusion of this study and some avenues for further research.

2 Background

2.1 Selectional Preference

Selectional preferences (p) are verb-sense specific. It is possible for a particular sense of a verb to have more than one selectional preference. A selectional preference of a verb-sense (s) refers to the predicate-argument structure relationship between s and its arguments. Formally, a selectional preference is a function whose domain is the finite set of semantic roles (r) and whose range is a finite set of noun types (t):

$$p(r_i) = t_j$$

For example, the first sense of the verb "eat" (eat_1: *take in solid food*) in WordNet (Miller, 1995) would have a selectional preference that requires the **subject** of the verb to be nouns of the **animate** type and the **object** of the verb to be nouns of the **food** type; whereas the fourth sense of "eat" (eat_4: *use up (resources or materials)*) would have a selectional preference which allows the subject of the verb to be of both animate type and inanimate type.

Since there does not exist a set of commonly accepted noun types, it is common for different selectional preference based WSD systems to invent their noun types.

One can draw a parallel between verb selectional preferences of natural language and function overloading of the programming language Java. In Java, two or more functions can be declared with the same name, each of these functions will have a different argument list which the Java interpreter uses at runtime to select (disambiguate) the correct function. The argument list of Java functions is an ordered list of Java object *types*. Similarly, one can treat the different senses of any verb as different functions sharing the same name, and distinguish between them based on which type of nouns are used in which semantic role of the verb.

2.2 Related Work

Resnik (1997) describes a WSD system which uses selectional preferences to train an entropy based probabilistic model classifier. Resnik defines the *prior* distribution $Pr_p(t)$ as the probability of the noun-type t occurring in a particular selectional preference p. From the prior distribution, Resnik defines the *selectional preference strength* of a particular verb sense s with respect to a particular selectional preference p over a finite set of noun types T as:

$$\begin{aligned} St_p(s) &= D(Pr_p(t|s) \| Pr_p(t)) \\ &= \sum_{t \in T} Pr_p(t|s) \log \frac{Pr_p(t|s)}{Pr_p(t)} \end{aligned}$$

From the above equation, it is obvious that the selectional preference strength of a verb sense s depends on how much mutual information the noun types of its arguments share. In other words, verb senses which take a small set of nouns as arguments are easier to disambiguate.

With the selection preference strength, Resnik further defined the *selectional association* value between a verb sense s and a noun-type t as:

$$A_p(t,s) = \frac{1}{St_p(s)} Pr_p(t|s) \log \frac{Pr_p(t|s)}{Pr_p(t)}$$

The disambiguation of a polysemous verb v using Resnik's system is therefore achieved in the following way: Suppose the noun n is an argument to a polysemous verb v; Let $[s_1, s_2, ..., s_n]$ be v's senses; let $[ns_1, ns_2, ..., ns_k]$ be n's senses; and for each ns_j, let H_j be the set of WordNet synsets which are hypernyms of ns_j; compute the following for each s_i:

$$VA(s_i) = \max_{ns_j \in H_j} A_p(s_i, ns_j)$$

Then the verb sense(s) which maximise(s) the function VA will be chosen as the most appropriate sense(s) for v. Since Resnik's system is trained and evaluated on WordNet, he used a subset of WordNet noun synsets as the noun-types of his selectional preferences. Therefore, each ns_i is a noun type.

I believe this method of choosing noun types is a weakness of Resnik's system. It is not clear from his description whether this subset of noun synsets were hand picked or computed from the available data. If these synsets were hand picked (which is the likely scenario), then the resulting system could suffer from poor coverage because it was highly unlikely that the hand picked set of noun types were complete or compatible with the WordNet noun hypernym hierarchy. To illustrate this

problem, consider the verb-object relationship between $drink_1$ (*take in liquids* and its objects: if the noun-type **beverage** ($beverage_1$) is chosen as a noun type (as it was in Resnik's paper), and the sentences "John drank wine" and "Joe drank coffee" are in the training data, since *coffee*$_1$ and *wine*$_1$ both have **beverage** as hypernym then the probability of $Pr(beverage|drink_1)$ is very likely to be high. However, if in the testing data, the system encounters the sentence "John drank some water", then because "water" does not have **beverage** as a hypernym in WordNet, it would be unlikely for the system to identify the correct sense of "drink".

On the other hand, if the subset of noun types are computed from the training data, then all the hypernyms of the nouns in the training data would also be taken into account in the estimation of $Pr_p(t|s)$. Furthermore, since the hypernym of a noun n would always describe a more general concept than n, then it is natural that the noun types describing the most general concepts would produce the highest value for the estimation of $Pr_p(t|s)$. However, the more general the noun type is, the less distinguishing feature it would be able to provide, therefore such noun types would not be effective for the WSD task.

Another weakness of Resnik's system is that the selectional preferences used in this system were constructed with only a single semantic role, e.g. the object of the verb or the subject of the verb. Therefore, these selectional preferences could only provide limited features useful for sense disambiguation.

3 Methodology

3.1 System Architecture

The system developed in this study takes semantic-role-labelled sentences as inputs and trains a classifier which can be used for the disambiguation of verbs.

The system consists of two major components: the selectional preference construction module and the classifier training and disambiguation module. When the system is given a semantic-role-labelled sentence, it first constructs the selectional preferences from the labelled semantic roles and their head nouns. These selectional preferences are then passed to the statistical classifier for the training or the disambiguation of the verb. In this study, two types of statistical classifiers were investigated: a Support Vector Machine (SVM) classifier and a Bayesian classifier.

A state-of-the-art semantic role labelling system, "ASSERT" (Pradhan et al., 2004), was used for the task of semantic role labelling. The influence of the ASSERT will be discussed in section 4. In the remainder of this section, I will give details of the two main modules of the WSD system.

3.2 Selectional Preference Construction

In this study, the WordNet **noun hypernym hierarchy** (NHH) is used to generate the noun-types used to construct the selectional preferences. A noun synset ns_a is the hypernym of another noun synset ns_b if ns_a denotes a more general concept than ns_b. A hypernym hierarchy for a noun synset ns is the tree structure which includes all the direct and indirect hypernyms of ns.

Each path from the most specific node in the NHH to the most general node is treated as a separate noun-type. For example, the NHH of the first sense of "apple" ($apple_1$: *fruit with red or yellow or green skin and sweet to tart crisp whitish flesh*) would generate the following noun-types:

t_1: ($entity_1$, $substance_1$, $solid_1$, $food_2$, $produce_1$, $edible\_fruit_1$, $apple_1$)

t_2: ($entity_1$, $object_1$, $natural\_object_1$, $plant\_part_1$, $plant\_organ_1$, $reproductive\_structure_1$, $fruit_1$, $edible\_fruit_1$, $apple_1$)

t_3: ($entity_1$, $object_1$, $natural\_object_1$, $plant\_part_1$, $plant\_organ_1$, $reproductive\_structure_1$, $fruit_1$, $pome_1$, $apple_1$)

There are two advantages of using paths extracted from WordNet NHH as the noun-types. First, it eliminates the need for a set of hand generated noun-types which is most likely to be not as comprehensive as WordNet. Second, since the noun-types are set of noun synsets of varying degrees of generality, it is possible to compute partial equality between them, this partial equality will then be applicable to the comparison between selectional preferences, thereby increasing the coverage and potentially the accuracy of the system.

To illustrate how selectional preferences are constructed from semantic-role-labelled sentences, suppose we have the following sentence:

E1 [The monkey]$_{arg0}$ [ate]$_{target}$ [an apple]$_{arg1}$

The head nouns for **arg0** and **arg1** are "monkey" and "apple" respectively. Since the system does not know which senses of these words are being used here, it will have to consider all senses of both words. In WordNet, "monkey" and "apple" correspond to the NHHs shown in figure 1.

Each **path** in the above NHHs is a noun-type. As we can see, there are 3 potential noun-types for "monkey" (**arg0**), and 4 potential noun-types for

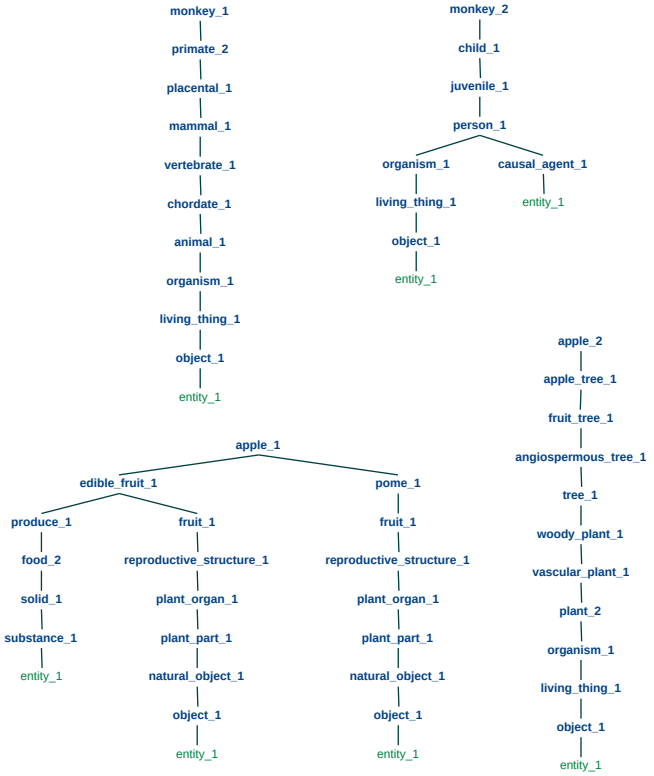

Figure 1: Example NHHs

"apple" (**arg1**). Therefore, the sentence E1 gives rise to 12 potential selectional preferences.

3.3 Training of the SVM Classifier

The SVM classifier is the first of the two classifiers investigated in this study. Since most verbs have more than two senses, the SVM classifier was trained to be multi-class, and each sense was treated as an independent class. Two types of multi-class classification were experimented: One-class-Against-the-Rest-of-the-classes and One-class-Against-One-other-class.

The attributes used in the classification are combinations of semantic role and WordNet noun synset. Recall that a selectional preference is a function mapping semantic roles to noun types; and each noun-type is a set of WordNet noun synsets. Each noun synset will be combined with its respective semantic role to form a feature. Therefore, if the total number of semantic roles is N_r and the total number of WordNet noun synsets is N_{ns}, the total number of dimensions or features is then $N_f = N_r \times N_{ns}$. During the training and the classification, all the selectional preferences generated for the **same** instance of a verb are used to create a single feature vector. If a synset ns_i appears in the noun-type for a particular semantic role r_j, then the feature corresponding to the (r_j, ns_i) tuple will have the value of 1.0, otherwise this feature will have the value of 0.0. Furthermore, all the selectional preference generated are always stored in the same feature vector. The total number of features may seem excessive, however, since the training data is unlikely to contain all the relevant selectional preferences, it is therefore necessary to include all the possible features during training and classification.

Two types of SVM kernels were experimented with in this study, linear and degree 2 polynomial.

3.4 Training of the Probabilistic Classifier

Since theoretically it is possible for a verb to take a large number of nouns for any of its semantic roles, the training of the probabilistic classifier would suffer from the data sparseness problem if no preprocessing is performed on the training data.

The preprocessing performed in this study is based on the theory of argument fusion (Jackendoff, 1990). Its main purpose is to extract common features from the noun types and give them appropriate mass in the probabilistic distribution. For example, suppose the training data consists of the following sentences for eat_1 (*take in solid food*).

S1 [The monkey]$_{arg0}$ [ate]$_{target}$ [an apple]$_{arg1}$

S2 John's dietitian allowed [him]$_{arg0}$ to [eat]$_{target}$ only [one slice of the $cake_3$]$_{arg1}$ at his birthday party.

S1 would generate the following selectional preferences:

S1_1 **arg0** $(entity_1, object_1, living\_thing_1, organism_1, animal_1, chordate_1, verebrate_1, mammal_1, placental_1, primate_2, monkey_1)$

arg1 $(entity_1, object_1, natural\_object_1, plant\_part_1, plant\_organ_1, reproduction\_structure_1, fruit_1, edible\_fruit_1, apple_1)$

S1_2 **arg0** $(entity_1, object_1, living\_thing_1, organism_1, animal_1, chordate_1, verebrate_1, mammal_1, placental_1, primate_2, monkey_1)$

arg1 $(entity_1, object_1, natural\_object_1, plant\_part_1, plant\_organ_1, reproduction\_structure_1, fruit_1, pome_1, apple_1)$

S1_3 **arg0** $(entity_1, object_1, living\_thing_1, organism_1, animal_1, chordate_1, verebrate_1, mammal_1, placental_1, primate_2, monkey_1)$

arg1 $(entity_1, substance_1, solid_1, food_2, produce_1, edible\_fruit_1, apple_1)$

S2 would generate the following selectional preferences:

S2_1 **arg0** $(entity_1, casual\_agent_1, person_1, male_2, man_1, John)$

arg1 $(entity_1, substance_1, solid_1, food_2 \; baked\_goods_1, cake_3)$

S2_2 **arg0** $(entity_1, object_1, living\_thing_1, organism_1, person_1, male_2, man_1, John)$

arg1 $(entity_1, substance_1, solid_1, food_2 \; baked\_goods_1, cake_3)$

It can be observed that some of these selectional preferences have partial overlappings among the noun-types of the same semantic roles. These overlappings capture what is in common between the examples from the training data. Intuitively, the overlappings are more suitable to be the selectional preferences than the individual training examples. For example, consider the selectional preferences generated by S1 and S2 for eat_1, one of the overlappings between them is:

S12_1 **arg0** $(entity_1, object_1, living\_thing_1, organism_1)$

arg1 $(entity_1, substance_1, solid_1, food_2)$

It is obvious that S12_1 captures almost exactly what eat_1's selectional preference really should be, namely that the subject of the verb has to be some living organism and the object of the verb has to be some kind of food. In the remainder of this paper, selectional preferences constructed through the process of argument fusion will be referred to as **fused selectional preferences**, and selectional preferences directly constructed from the training data will be referred to as **raw selectional preferences**. Since fused selectional preferences are more prototypical than the raw ones, it would make sense to give them greater mass in the final probability distribution.

Formally, the frequency of the selectional preferences are estimated in the following way:

Let rp_i be a raw selectional preference, its frequency ($C(rp_i)$) is the number of times rp_i appears in the training examples.

Let fp_j be a fused selectional preference, and let $[rp_1, rp_2, ..., rp_k]$ be the set of raw selectional preferences from which fp_j was derived, then fp_j's frequency is calculated as:

$$C(fp_j) = \sum_{i=1}^{k} C(rp_i)$$

Similarly, the conditional frequency of the selectional preference p_i given the verb sense s_j ($C(p_i|s_j)$) is estimated as the number of times p_i co-occurs with s_j.

The two frequency distributions are then used to construct the corresponding probability distributions which are then smoothed to allow far unseen data.

The classification of previously unseen data is not as simple as finding the verb sense s_i which maximises the probability of $Pr(s_i|p_j)$. Firstly, let P^c be the set of candidate selectional preferences $[p_1^c, ..., p_n^c]$ extracted with respect to an ambiguous verb v in a given context. Let P^t be the set of selectional preferences $[p_1^t, ..., p_m^t]$ from the training data. Suppose the set of senses of v is $S = [s_1, ..., s_k]$, then the most suitable sense(s) of v will be chosen in the following equation:

$$s_{max} = \underset{s_i \in S, p_j^c \in P^c}{\operatorname{argmax}} (max(Pr(s_i|p_j^c))$$

From Bayes' rule, $Pr(s_i|p_j^c)$ is calculated as follows:

$$Pr(s_i|p_j^c) = \frac{Pr(p_j^c|s_i)Pr(s_i)}{Pr(p_j^c)}$$

However, since it is very likely that p_j^c has not previously been seen in the training data, $Pr(s_i|p_j^c)$ is therefore estimated as follows:

$$Pr(s_i|p_j^c) = \max_{p_k^t \in P^t} (Pr(s_i|p_k^t) \cdot sim(p_j^c, p_k^t))$$

The function $sim(p_a, p_b)$ calculates the similarity between two given selectional preferences. Let $dom(p_a)$ and $dom(p_b)$ be the sets of semantic roles applicable to p_a and p_b respectively. Recall that in this study, a noun-type t is a **set** of WordNet noun synsets, then the function sim works in the following way:

$$sim(p_a, p_b) = \begin{cases} 0 & dom(p_a) \neq dom(p_b) \\ \sum_{r_i \in dom(p_a)} \cos(p_a(r_i), p_b(r_i)) & otherwise \end{cases}$$

4 Results

The system developed in this study was evaluated using the Semcor2.0 data and the Propbank data. Two types of baseline performances were used in the evaluation: the majority sense baseline (baseline 1) and the bag-of-synsets baseline (baseline 2).

The bag-of-synsets baseline works by first collecting the 10 nouns closest to and before the verb,

and the 10 nouns closest to and after the verb; then extracting the synsets from their WordNet noun hypernym hierarchy; and finally using these synsets as features to training a support vector machine classifier. The purpose of this baseline is to see whether the additional information provided by the semantic roles can indeed improve the performance of WSD.

Because of the diverse natural of verb selectional preferences and the different availabilities of the verb specific training data, the evaluation of the two classifiers was performed in a verb-by-verb fashion. The verbs selected for evaluation are: "bear", "eat", "kick", "look", "run", and "serve". As shown in table 1, these verbs are chosen because they represent a variety of transitivities, semantic role combinations, and different degrees of similarities between the senses. The senses of these verbs are defined in WordNet2.0.

Verb	Intran.[1]	Trans.[2]	Compl.[3]	NSR [4]
bear	no	yes	no	11
eat	yes	yes	no	2
kick	yes	yes	no	9
look	yes	no	yes	11
run	yes	yes	no	28
serve	no	yes	yes	9

Table 1: Semantic Properties of the verbs

The following classifiers were trained and evaluated:

C1 SVM classifier with a linear kernel

C2 SVM classifier with a degree 2 polynomial kernel

C3 Naive Bayes classifier using thematic role tag set

Table 2 shows the number of senses and the majority class baselines of the above verbs:

Verb	Majority Baseline 1	No. of senses
bear	31.58%	9
eat	76.27%	3
kick	45%	3
look	56.9%	8
run	33.75%	26
serve	27.81%	11

Table 2: Majority class baseline

[1]Intransitive

[2]Transitive

[3]Require Prepositional Complement

[4]Number of applicable semantic roles according to Propbank

Tables 3 to 5 show the results (Accuracy) of the above classifiers trained on 30%, 50%, and 80% of the training data:

Verbs	baseline 2	C1	C2	C3
bear	30.23	37.9	31.62	20.23
eat	75	61.5	64.5	70.25
kick	43.75	42.5	46.25	43.16
look	5.97	49.25	50.14	26.27
run	4.31	4.83	3.97	5.69
serve	12.5	36.5	38	16.58

Table 3: Classifiers accuracy(%) when 30% data was used in the training

Verbs	baseline 2	C1	C2	C3
bear	6.67	41.33	37.33	20.33
eat	7.14	60.36	63.57	65.36
kick	18.18	54.54	50.9	45.45
look	57.63	50.69	52.5	34.79
run	1.19	5	9.52	6.91
serve	12.79	4.88	4.88	14.42

Table 4: Classifiers accuracy(%) when 50% data was used in the training

Verbs	baseline 2	C1	C2	C3
bear	14.29	40.71	44.28	24.29
eat	8.33	58.33	63.33	63.33
kick	20	35	56	40
look	33.89	47.29	48.81	39.32
run	2.56	4.61	7.17	5.64
serve	7.89	8.95	7.89	13.16

Table 5: Classifiers accuracy(%) when 80% data was used in the training

The most significant feature of the results is that the three classifiers all performed below the majority class baseline. These poor results were caused by a combination of the following factors: complex sentence, poor semantic role labelling, inconsistent data, too finely defined verb senses and inadequate smoothing of the probability distributions.

The sentences used in the evaluation are generally longer than 20 words and contain embedded clauses, metaphors and ellipses. For instance, one of the examples for "eat" (eat_1) is the sentence: *"The dialogue is sharp witty and candid typical don't* **eat** *the daisies material which has stamped the author throughout her books and plays and it was obvious that the Theatre-by-the-Sea audience liked it"*. In this sentence, there is no subject/AGENT for "eat". Another example for "eat" (eat_3) is:

"No_matter that it is his troops who rape Western women and eat Western men". In this sentence, "eat" is clearly used in a metaphoric way therefore should not be interpreted literally. These complex sentences not only increase the amount of noise in the data, but also make semantic role labelling difficult. According to my estimation, less than 30% of the sentences were correctly tagged with semantic roles.

Another problem with the semantic role labelling is that it only labels noun phrases. The impact of this problem is shown by the very poor result on the verb "serve" most of whose senses require either a propositional phrase or a verb phrase as compliment. For example, the sentence "The tree stump serves as a table" is annotated as "[The tree stump]$_{agent}$ [serves]$_{target}$ as [a]$_{proposition}$ table" which is clearly wrong.

The problem caused by the excessively fine-grained senses is that these senses have very similar (sometimes identical) selectional preferences which cause inconsistency in the training data. Take eat for example, the definitions of its first and second senses are: *"take in solid food"*, and *"eat a meal; take a meal"* respectively. In the training data, in *" She was personally sloppy, and when she had colds would blow her nose in the same handkerchief all day and keep it soaking wet dangling from her waist and when she gardened she would* **eat** *dinner with dirt on her calves"*, *eat* is labelled as having the first sense, but it is labelled as having the second sense in *"Charlie* **ate** *some supper in the kitchen and went_into the TV_room to hear the news."*. This type of inconsistency causes the classifiers to sometimes behave almost randomly with respect to the relevant senses.

The problem caused by the inadequate smoothing of the probability distributions is more subtle. Given a very frequent senses s_a and a very infrequent verb sense s_b and a candidate selectional preference p_i^c , the conditional probabilities of $Pr(s_a|p_i^c)$ and $Pr(s_b|p_i^c)$ depends on the values of $Pr(p_i^c|s_a) \cdot Pr(s_a)$ and $Pr(p_i^c|s_b) \cdot Pr(s_b)$. It is often the case that there are so many selectional preferences applicable to s_a that $\frac{Pr(p_i^c|s_a)}{Pr(p_i^c|s_b)} < \frac{Pr(s_b)}{Pr(s_a)}$, thereby making the Bayes classifier assign s_b to instances of s_a. Currently, the Lidstone probability distribution with γ of 0.0001 is used by the Bayes classifier; further study is required to select a more suitable probability distribution.

Another interesting feature of the results is that the differences of the accuracy is relatively small with respect to the different amounts of data used in training. This feature is expected because one of the assumptions made by selectional preference based WSD is that each semantic role of any verb sense should be filled by nouns of similar type, in other words, nouns that have something in common. Therefore even though the amount of training data is different, the common features between the nouns of the same semantic role can still be captured and used for disambiguation.

Finally, the results also show that verbs with higher order of transitivity are easier to disambiguate. This is also not surprising because higher transitivity means more semantic roles which in turn provides more disambiguating features.

5 Conclusion and Future work

This paper has presented a study of whether the performance of selectional preference based WSD could be improved by using the current state-of-art semantic role labelling system. Although very little performance improvement was able to be achieved by the systems developed in this study, a few useful observations could be made.

First, selectional preference based WSD systems do not require a large amount of training data, as demonstrated by the previous section. Therefore, they may be more useful or more effective than corpus based WSD systems when the amount of training data is very limited or to act as a bootstrapping mechanism.

Second, to a very large degree, the performance of a selectional preference based WSD system depends on how finely the different senses of a verb are defined and the total number of semantic roles associated with the senses. As demonstrated by the "eat" example, the finer the senses are defined, the less effective selectional preference will be.

Third, the performance of selectional preference based WSD systems is heavily influenced by the quality of the semantic role identification. More importantly, it is not sufficient to only use semantic roles which can only be filled by noun phrases, as "serve" illustrated in the previous section; prepositions and verbal complements are also likely to be useful to selectional preference based WSD systems.

The results of this study also merit several further research topics. First, the focus of this research was on the disambiguation of verbs. However, the results of the disambiguation also contains the sense information of the nouns which are the arguments to the disambiguated verbs. Therefore, the next step of the current research is to assess how well the system developed in this work would perform on the nouns.

A further extension to the current WSD system

would be to incorporate extra information such as the prepositions and other open class words in the disambiguation. This extension may require a hybrid WSD system incorporating selectional preference based mechanisms and corpus based mechanisms.

Finally, as it was observed that the performance of a selectional preference based WSD system was heavily influenced by the quality of semantic role labelling; it might also be possible to use selectional preference as a crude measure of the performances of semantic role labelling systems on unlabelled data. This is because it is likely for a particular semantic role to be filled by nouns of similar types, therefore nouns correctly labelled for the same semantic role should exhibit a greater similarity than if incorrectly labelled.

Acknowledgements

I would like thank my supervisors, Professor Steven Bird and Dr. Adrian Pearce, for reviewing the paper. This research was sponsored by Australian Postgraduate Awards (APA) from the Australian Research Council (ARC).

References

Ray Jackendoff, 1990. *Semantic Structures*, chapter 2. The MIT Press.

Daniel Jurafsky and James H. Martin, 2000. *Speech and Language Processing*, chapter 17. Prentice Hall, 1 edition.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM Press.

Diana McCarthy and John Carroll. 2003. Disambiguating nouns, verbs, and adjectives using automatically acquired selectional preferences. *Computational Linguistics*, 29(4):639–654, December.

George A. Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. 2004. Shallow semantic parsing using support vector machines. In *Proceedings of the Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL-2004), Boston, MA*, May 2–7.

Philip Resnik. 1997. Selectional preference and sense disambiguation. In *Proceedings of ACL Siglex Workshop on Tagging Text with Lexical Semantics, Why, What and How?, Washington, April 4-5, 1997.*

David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Meeting of the Association for Computational Linguistics*, pages 189–196.