# SRL-based Verb Selection for ESL

**[1,2]Xiaohua Liu, [3]Bo Han[*], [4]Kuan Li[*], [5]Stephan Hyeonjun Stiller and [2]Ming Zhou**
[1]School of Computer Science and Technology
Harbin Institute of Technology
[2]Microsoft Research Asia
[3]Department of Computer Science and Software Engineering
The University of Melbourne
[4]College of Computer Science
Chongqing University
[5]Computer Science Department
Stanford University
{xiaoliu, mingzhou, v-kuli}@microsoft.com
b.han@pgrad.unimelb.edu.au
sstiller@stanford.edu

## Abstract

In this paper we develop an approach to tackle the problem of verb selection for learners of English as a second language (ESL) by using features from the output of Semantic Role Labeling (SRL). Unlike existing approaches to verb selection that use local features such as n-grams, our approach exploits semantic features which explicitly model the usage context of the verb. The verb choice highly depends on its usage context which is not consistently captured by local features. We then combine these semantic features with other local features under the generalized perceptron learning framework. Experiments on both in-domain and out-of-domain corpora show that our approach outperforms the baseline and achieves state-of-the-art performance.

## 1 Introduction

Verbs in English convey actions or states of being. In addition, they also communicate sentiments and imply circumstances, e.g., in "*He got [gained] the scholarship after three interviews.*", the verb "*gained*" may indicate that the "*scholarship*" was competitive and required the agent's efforts; in contrast, "*got*" sounds neutral and less descriptive. Since verbs carry multiple important functions, misusing them can be misleading, e.g., the native speaker could be confused when reading "*I like looking [reading] books*". Unfortunately, according to (Gui and Yang, 2002; Yi et al., 2008), more than 30% of the errors in the Chinese Learner English Corpus (CLEC) are verb choice errors. Hence, it is useful to develop an approach to automatically detect and correct verb selection errors made by ESL learners.

However, verb selection is a challenging task because verbs often exhibit a variety of usages and each usage depends on a particular context, which can hardly be adequately described by conventional n-gram features. For instance, both "*made*" and "*received*" can complete "*I have __ a telephone call.*", where the usage context can be represented as "*made/received a telephone call*"; however, in "*I have __ a telephone call from my boss*", the prepositional phrase "*from my boss"* becomes a critical part of the context, which now cannot be described by n-gram features, resulting in only "*received*" being suitable.

Some researchers (Tetreault and Chodorow, 2008) exploited syntactic information and n-gram features to represent verb usage context. Yi et al. (2008) introduced an unsupervised web-based proofing method for correcting verb-noun collocation errors. Brockett et al. (2006) employed phrasal Statistical Machine Translation (SMT) techniques to correct countability errors. None of their methods incorporated semantic information.

[*] This work has been done while the author was visiting Microsoft Research Asia.



*Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1068–1076,
MIT, Massachusetts, USA, 9-11 October 2010. 

Unlike the other papers, we derive features from the output of an SRL (Màrquez, 2009) system to explicitly model verb usage context. SRL is generally understood as the task of identifying the arguments of a given verb and assigning them semantic labels describing the roles they play. For example, given a sentence "*I want to watch TV tonight*" and the target predicate "*watch*", the output of SRL will be something like "*I [A0] want to watch [target predicate] TV [A1] tonight [AM-TMP].*", meaning that the action "*watch*" is conducted by the agent "*I*", on the patient "*TV*", and the action happens "*tonight*".

We believe that SRL results are excellent features for characterizing verb usage context for three reasons: (i) Intuitively, the predicate-argument structures generated by SRL systems capture major relationships between a verb and its contextual participants and consequently largely determine whether or not the verb usage is proper. For example, in "*I want to watch a match tonight.*", "*match*" is the patient of "*watch*", and "*watch ... match*" forms a collocation, suggesting "*watch"* is appropriately used. (ii) Predicate-argument structures abstract away syntactic differences in sentences with similar meanings, and therefore can potentially filter out lots of noise from the usage context. For example, consider "*I want to watch a football match on TV tonight*": if "*match*" is successfully identified as the agent of "*watch*", "*watch ... football*", which is unrelated to the usage of "*watch*" in this case, can be easily excluded from the usage context. (iii) Research on SRL has made great achievements, including human-annotated training corpora and state-of-the-art systems, which can be directly leveraged.

Taking an English sentence as input, our method first generates correction candidates by replacing each verb with verbs in its pre-defined confusion set; then for every candidate, it extracts SRL-derived features; finally our method scores every candidate using a linear function trained by the generalized perceptron learning algorithm (Collins, 2002) and selects the best candidate as output.

Experimental results show that SRL-derived features are effective in verb selection, but we also observe that noise in SRL output adversely increases feature space dimensions and the number of false suggestions. To alleviate this issue, we use local features, e.g., n-gram-related features, and achieve state-of-the-art performance when all features are integrated.

Our contributions can be summarized as follows:

1. We propose to exploit SRL-derived features to explicitly model verb usage context.
2. We propose to use the generalized perceptron framework to integrate SRL-derived (and other) features and achieve state-of-the-art performance on both in-domain and out-of-domain test sets.

Our paper is organized as follows: In the next section, we introduce related work. In Section 3, we describe our method. Experimental results and analysis on both in-domain and out-of-domain corpora are presented in Section 4. Finally, we conclude our paper with a discussion of future work in Section 5.

## 2 Related Work

SRL results are used in various tasks. Moldovan et al. (2004) classify the semantic relations of noun phrases based on SRL. Ye and Baldwin (2006) apply semantic role–related information to verb sense disambiguation. Narayanan and Harabagiu (2004) use semantic role structures for question answering. Surdeanu et al. (2003) employ predicate-argument structures for information extraction.

However, in the context of ESL error detection and correction, little study has been carried out on clearly exploiting semantic information. Brockett et al. (2006) propose the use of the phrasal statistical machine translation (SMT) technique to identify and correct ESL errors. They devise several heuristic rules to generate synthetic data from a high-quality newswire corpus and then use the synthetic data together with their original counterparts for SMT training. The SMT approach on the artificial data set achieves encouraging results for correcting countability errors. Yi et al. (2008) use web frequency counts to identify and correct determiner and verb-noun collocation errors. Compared with these methods, our approach explicitly models verb usage context by leveraging the SRL output. The SRL-based semantic features are integrated, along with the local features, into the generalized perceptron model.

## 3 Our Approach

Our method can be regarded as a pipeline consisting of three steps. Given as input an English sentence written by ESL learners, the system first checks every verb and generates correction candidates by replacing each verb with its confusion set. Then a feature vector that represents verb usage context is derived from the outputs of an SRL system and then multiplied with the feature weight vector trained by the generalized perceptron. Finally, the candidate with the highest score is selected as the output.

### 3.1 Formulation

We formulate the task as a process of generating and then selecting correction candidates:

$$s^* = \arg\max_{s \in GEN(s')} Score(s) \quad (1)$$

Here $s'$ denotes the input sentence for proofing, $GEN(s')$ is the set of correction candidates, and $Score(s)$ is the linear model trained by the perceptron learning algorithm, which will be discussed in section 3.4.

We call every target verb in $s'$ a *checkpoint*. For example, "*sees*" is a checkpoint in "*Jane **sees** TV every day.*". Correction candidates are generated by replacing each checkpoint with its confusions. Table 1 shows a sentence with one checkpoint and the corresponding correction candidates.

| Input | *Jane **sees** TV every day.* |
|---|---|
| Candidates | *Jane **watches** TV every day.* |
| | *Jane **looks** TV every day.* |
| | … |

Table 1. Correction candidate list.

One state-of-the-art SRL system (Riedel and Meza-Ruiz, 2008) is then utilized to extract predicate-argument structures for each verb in the input, as illustrated in Table 2.

Semantic features are generated by combining the predicate with each of its arguments; e.g., "*watches_A0_Jane*", "*sees_A0_Jane*", "*watches_A1_TV*" and "*sees_A1_TV*" are semantic features derived from the semantic roles listed in Table 2.

| Sentence | Semantic roles |
|---|---|
| *Jane **sees** TV every day* | *Predicate: sees;*<br>*A0: Jane;*<br>*A1: TV;* |
| *Jane **watches** TV every day* | *Predicate: watches;*<br>*A0: Jane;*<br>*A1: TV;* |

Table 2. Examples of SRL outputs.

At the training stage, each sentence is labeled by the SRL system. Each correction candidate $s$ is represented as a feature vector $\Phi(s) \in R^d$, where $d$ is the total number of features. The feature weight vector is denoted as $\vec{w} \in R^d$, and $Score(s)$ is computed as follows:

$$Score(s) = \Phi(s) \cdot \vec{w} \quad (2)$$

Finally, $Score(s)$ is applied to each candidate, and $s^*$, the one with the highest score, is selected as the output, as shown in Table 3.

| | Correction candidate | Score |
|---|---|---|
| $s^*$ | *Jane **watches** TV every day.* | 10.8 |
| | *Jane **looks** TV every day.* | 0.8 |
| | *Jane **reads** TV every day.* | 0.2 |
| | ... | … |

Table 3. Correction candidate scoring.

In the above framework, the basic idea is to generate correction candidates with the help of predefined confusion sets and apply the global linear model to each candidate to compute the degree of its fitness to the usage context that is represented as features derived from SRL results.

To make our idea practical, we need to solve the following three subtasks: (i) generating the confusion set that includes possible replacements for a given verb; (ii) representing the context with semantic features and other complementary features; and (iii) training the feature weight. We will describe our solutions to those subtasks in the rest of this section.

| *I* | *have* | ***made[opened]*** | *an* | *American* | *bank* | *account* | *in* | *Boston* | . |
|---|---|---|---|---|---|---|---|---|---|
| [A0] | | [Predicate] | | | | [A1] | [AM-LOC] | | |

Table 4. An example of SRL output.

### 3.2 Generation of Verb Confusion Sets

Verb confusion sets are used to generate correction candidates. Due to the great number of verbs and their diversified usages, manually collecting all verb confusions in all scenarios is prohibitively time-consuming. To focus on the study of the effectiveness of semantic role features, we restrict our research scope to correcting verb selection errors made by Chinese ESL learners and select fifty representative verbs which are among the most frequent ones and account for more than 50% of ESL verb errors in the CLEC data set. For every selected verb we manually compile a confusion set using the following data sources:

1. Encarta treasures. We extract all the synonyms of verbs from the Microsoft Encarta Dictionary, and this forms the major source for our confusion sets.

2. English-Chinese Dictionaries. ESL learners may get interference from their mother tongue (Liu et al., 2000). For example, some Chinese people mistakenly say "see newspaper", partially because the translation of "see" co-occurs with "newspaper" in Chinese. Therefore English verbs in the dictionary sharing more than two Chinese meanings are collected. For example, "*see*" and "*read*" are in a confusion set because they share the meanings of both "看" ("*to see*", "*to read*") and "领会" ("*to grasp*") in Chinese.

3. An SMT translation table. We extract paraphrasing verb expressions from a phrasal SMT translation table learnt from parallel corpora (Och and Ney, 2004). This may help us use the implicit semantics of verbs that SMT can capture but a dictionary cannot, such as the fact that the verb

Note that verbs in any confusion set that we are not interested in are dropped, and that the verb itself is included in its own confusion set. We leave it to our future work to automatically construct verb confusions.

### 3.3 Verb Usage Context Features

The verb usage context[1] refers to its surrounding text, which influences the way one understands the expression. Intuitively, verb usage context can take the form of a collocation, e.g., "*watch … TV*" in "*I saw [watched] TV yesterday.*" ; it can also simply be idioms, e.g., we say "*kick one's habit*" instead of "*remove one's habit*".

We use features derived from the SRL output to represent verb usage context. The SRL system accepts a sentence as input and outputs all arguments and the semantic roles they play for every verb in the sentence. For instance, given the sentence "*I have opened an American bank account in Boston.*" and the predicate "*opened*", the output of SRL is listed in Table 4, where *A0* and *A1* are two core roles, representing the agent and patient of an action, respectively, and other roles starting with "*AM-*"are adjunct roles, e.g., *AM-LOC* indicates the location of an action. Predicate-argument structures keep the key participants of a given verb while dropping other unrelated words from its usage context. For instance, in "*My teacher said Chinese is not easy to learn.*", the SRL system recognizes that "*Chinese*" is not the *A1*-argument of "*said*". So "*say _ Chinese*", which is irrelevant to the usage of *said*, is not extracted as a feature.

The SRL system, however, may output erroneous predicate-argument structures, which negatively affect the performance of verb selection. For instance, for the sentence "*He hasn't done anything but take [make] a lot of money*", "*lot*" is incorrectly identified as the patient of "*take*", making it hard to select "*make*" as the proper verb even though "*make money*" forms a sound collocation. To tackle this issue, we use local textual features, namely features related to n-gram, chunk and chunk headword, as shown in Table 5. Back-off features are generated by replacing the word with its POS tag to alleviate data sparseness.

[1] http://en.wikipedia.org/wiki/Context_(language_use)

| Local: trigrams |
|---|
| *have_**opened*** |
| *have_**opened**_a* |
| ***opened**_an_American* |
| *PRP_VBP_**opened*** |
| *VBP_**opened**_DT* |
| ***opened**_DT_JJ* |
| Local: chunk |
| *have_**opened*** |
| ***opened**_an_American_investment_bank_account* |
| *PRP_**opened*** |
| ***opened**_NN* |
| Semantic: SRL derived features |
| *A0_I_**opened*** |
| ***opened**_A1_account* |
| ***opened**_AM-LOC_in* |
| **...** |

Table 5. An example of feature set.

### 3.4 Perceptron Learning

We choose the generalized perceptron algorithm as our training method because of its easy implementation and its capability of incorporating various features. However, there are still two concerns about this perceptron learning approach: its ineffectiveness in dealing with inseparable samples and its ignorance of weight normalization that potentially limits its ability to generalize. In section 4.4 we show that the training error rate drops significantly to a very low level after several rounds of training, suggesting that the correct candidates can almost be separated from others. We also observe that our method performs well on an out-of-domain test corpus, indicating the good generalization ability of this method. We leave it to our future work to replace perceptron learning with other models like Support Vector Machines (Vapnik, 1995).

In Figure 1, $s_i$ is the *i*th correct sentence within the training data. *T* and *N* represent the number of training iterations and training examples, respectively. $GEN(s^i)$ is the function that outputs all the possible corrections for the input sentence $s^i$ with each *checkpoint* substituted by one of its confusions, as described in Section 3.1. We observe that the generated candidates sometimes contain reasonable outputs for the verb selection task, which should be removed. For instance, in "*... reporters could not take [make] notes or tape the conversation*", both "*take*" and "*make*" are suitable verbs in this context. To fix this issue, we trained a trigram language model using SRILM (Stolcke, 2002) on LDC data[2] , and calculated the logarithms of the language model score for the original sentence and its artificial manipulations. We only kept manipulations with a language model score that is *t* lower than that of the original sentence. We experimentally set $t = 5$.

**Inputs**: training examples $s^i$, $i=1...N$
**Initialization**: $\vec{w} = 0$
**Algorithm**:
For $r= 1..\ T$, $i= 1..N$
Calculate $o = \arg\max_{s=Gen(s^i)} \Phi(s) \cdot \vec{w}$
If $s^i \neq o$
$\vec{w} = \vec{w} + \Phi(s^o) - \Phi(o)$
**Outputs**: $\vec{w}$

Figure 1. The perceptron algorithm, adapted from Collins (2002).

$\Phi$ in Figure 1 is the feature extraction function. $\Phi(o)$ and $\Phi(s^i)$ are vectors extracted from the output and oracle, respectively. A vector field is filled with 1 if the corresponding feature exists, or 0 otherwise; $\vec{w}$ is the feature weight vector, where positive elements suggest that the corresponding features support the hypothesis that the candidate is correct.

The training process is to update $\vec{w}$, when the output differs from the oracle. For example, when *o* is "*I want to **look** TV*" and $s^i$ is "*I want to **watch** TV*", $\vec{w}$ will be updated.

We use the averaged Perceptron algorithm (Collins, 2002) to alleviate overfitting on the training data. The averaged perceptron weight vector is defined as

$$\vec{\gamma} = \frac{1}{TN} \sum_{i=1..N, r=1..T} \vec{w}^{i,r} \qquad (3)$$

where $\vec{w}^{i,r}$ is the weight vector immediately after the *i*th sentence in the *r*th iteration.

[2] http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp ?catalogId=LDC2005T12

## 4 Experiments

In this section, we compare our approach with the SMT-based approach. Furthermore, we study the contribution of predicate-argument-related features, and the performances on verbs with varying distance to their arguments.

### 4.1 Experiment Preparation

The training corpus for perceptron learning was taken from LDC2005T12. We randomly selected newswires containing target verbs from the *New York Times* as the training data. We then used the *OpenNLP* package[3] to extract sentences from the newswire text and to parse them into the corresponding tokens, POS tags, and chunks. The SRL system is built according to Riedel and Meza-Ruiz (2008), using the CoNLL-2008 shared task data for training. We assume that the newswire data is of high quality and free of linguistic errors, and finally we gathered 20000 sentences that contain any of the target verbs we were focusing on. We experimentally set the number of training rounds to $T = 50$.

We constructed two sets of testing data for in-domain and out-of-domain test purposes, respectively. To construct the in-domain test data, we first collected all the sentences that contain any of the verbs we were interested in from the previous unused LDC dataset; then we replaced any target verb in our list with a verb in its confusion set; next, we used the language-model-based pruning strategy described in 3.4 to drop possibly correct manipulations from the test data; and finally we randomly sampled 5000 sentences for testing.

To build the out-of-domain test dataset, we gathered 186 samples that contained errors related to the verbs we were interested in from English blogs written by Chinese and from the CLEC corpus, which were then corrected by an English native speaker. Furthermore, for every error involving the verbs in our target list, both the verb and the word that determines the error are marked by the English native speaker.

### 4.2 Baseline

We built up a phrasal SMT system with the word re-ordering feature disabled, since our task only concerns the substitution of the target verb. To construct the training corpus, we followed the idea in Brockett et al. (2006), and applied a similar strategy described in section 3.4 to the SRL system's training data to generate aligned pairs.

### 4.3 Evaluation Metric

We employed the following metrics adapted from (Yi et al., 2008): *revised precision (RP), recall of the correction (RC)* and *false alarm (FA).*

$$RP = \frac{\#\ of\ Correct\ Proofings}{\#\ of\ All\ Checkpoints} \quad (4)$$

*RP* reflects how many outputs are correct usages. The output is regarded as a correct suggestion if and only if it is exactly the same as the answer. Paraphrasing scenarios, for example, the case that the output is "*take notes*" and the answer is "*make notes*", are counted as errors.

$$RC = \frac{\#\ of\ Correct\ Modified\ Proofings}{\#\ of\ Total\ Errors} \quad (5)$$

*RC* indicates how many erroneous sentences are corrected among all the errors. It measures the system's coverage of verb selection errors.

$$FA = \frac{\#\ of\ Incorrect\ Modified\ Checkpoints}{\#\ of\ All\ Checkpoints} \quad (6)$$

FA is related to the cases where a correct verb is mistakenly replaced by an inappropriate one. These false suggestions are likely to disturb or even annoy users, and thus should be avoided as much as possible.

### 4.4 Results and Analysis

The training error curves of perceptron learning with different feature sets are shown in Figure 2. They drop to a low error rate and then stabilize after a few number of training rounds, indicating that most of the cases are linearly separable and that perceptron learning is applicable to the verb selection task.

We conducted feature selection by dropping features that occur less than $N$ times. Here $N$ was experimentally set to 5. We observe that, after feature selection, some useful features such as "*watch_A1_TV*" and "*see_A1_TV*" were kept, but some noisy features like *"Jane_A0_sees"* and *"Jane_A0_watches"* were removed, suggesting the effectiveness of this feature selection approach.

[3] http://opennlp.sourceforge.net/

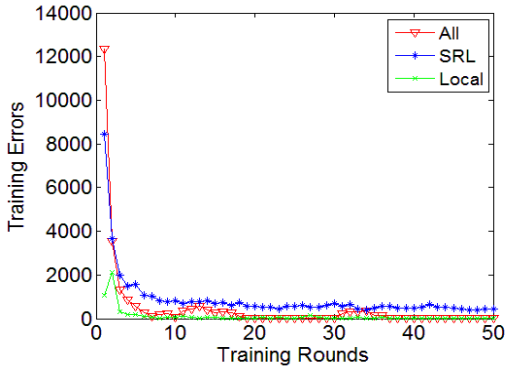


Figure 2. Training error curves of the perceptron.

We tested the baseline and our approach on the in-domain and out-of-domain corpora. The results are shown in Table 7 and 8, respectively.

In the in-domain test, the SMT-based approach has the highest false alarm rate, though its output with word insertions or deletions is not considered wrong if the substituted verb is correct. Our approach, regardless of what feature sets are used, outperforms the SMT-based approach in terms of all metrics, showing the effectiveness of perceptron learning for the verb selection task. Under the perceptron learning framework, we can see that the system using only SRL-related features has higher revised precision and recall of correction, but also a slightly higher false alarm rate than the system based on only local features. When local features and SRL-derived features are integrated together, the state-of-the-art performance is achieved with a 5% increase in recall, and minor changes in precision and false alarm.

In the out-of-domain test, the SMT-based approach performs much better than in the in-domain test, especially in terms of false alarm rate, indicating the SMT-based approach may favor short sentences. However, its recall drops greatly. We observe similar performance differences between the systems with different feature sets under the same perceptron learning framework, reaffirming the usefulness of the SRL-based features for verb selection.

We also conducted significance test. The results confirm that the improvements (SRL+Local vs. SMT-based) are statistically significant (p-value < 0.001) for both the open-domain and the in-domain experiments.

Furthermore, we studied the performance of our system on verbs with varying distance to their arguments on the out-of-domain test corpus.

| Local | d<=2 | 2<d<=4 | d>4 |
|---|---|---|---|
| RP | 64.3% | 60.3% | 59.4% |
| RC | 34.6% | 33.1% | 28.9% |
| FA | 3.0% | 6.3% | 5.0% |
| SRL | d<=2 | 2<d<=4 | d>4 |
| RP | 65.1% | 60.1% | 62.1% |
| RC | 40.3% | 34.0% | 36.9% |
| FA | 5.0% | 6.7% | 6.3% |

Table 9. Performance on verbs with different distance to their arguments on out-of-domain test data.

Table 9 shows that the system with only SRL-derived features performs significantly better than the system with only local features on the verb whose usage depends on a distant argument, i.e., one where the number of words between the predicate and the argument is larger than 4. To understand the reason, consider the following sentence:

"*It's raining outside. Please wear[take] the black raincoat with you.*"

| | SMT-based | Our method | | |
|---|---|---|---|---|
| | | SRL | Local | SRL + Local |
| RP | 48.4% | 64.5% | 62.2% | 66.4% |
| RC | 23.5% | 40.2% | 32.9% | 46.4% |
| FA | 13.3% | 5.6% | 4.2% | 6.8% |

Table 7. In-domain test results.

| | SMT-based | Our method | | |
|---|---|---|---|---|
| | | SRL | Local | SRL + Local |
| RP | 50.7% | 64.0% | 62.6% | 65.5% |
| RC | 13.5% | 39.0% | 33.3% | 44.0% |
| FA | 6.1% | 5.5% | 4.0% | 6.5% |

Table 8. Out-of-domain test results.

Intuitively, "*wear*" and "*take*" seem to fill the blank well, since they both form a collocation with "*raincoat*"*;* however, when *"with [AM-MNR] you*" is considered as part of the context, "*wear*" no longer fits it and "*take*" wins. In this case, the long-distance feature devised from *AM-MNR* helps select the suitable verb, while the trigram features cannot because they cannot represent the long distance verb usage context.

We also find some typical cases that are beyond the reach of the SRL-derived features. For instance, consider "*Everyone doubts [suspects] that Tom is a spy.*". Both of the verbs can be followed by a clause. However, the SRL system regards "*is*", the predicate of the clause, as the patient, resulting in features like "*doubt_A1_is*" and "*suspect_A1_is*", which capture nothing about verb usage context. However, if we consider the whole clause "*suspect_Tom is a spy*" as the patient, this could result in a very sparse feature that would be filtered. In the future, we will combine word-level and phrase-level SRL systems to address this problem.

Besides its incapability of handling verb selection errors involving clauses, the SRL-derived features fail to work when verb selection depends on deep meanings that cannot be captured by current shallow predicate-argument structures. For example, in "*He was wandering in the park, spending [killing] his time watching the children playing.*", though "*spending*" and "*killing*" fit the syntactic structure and collocation agreement, and express the meaning "*to allocate some time doing something*", the word "*wandering*" suggests that "*killing*" may be more appropriate. Current SRL systems cannot represent the semantic connection between two predicates and thus are helpless for this case. We argue that the performance of our system can be improved along with the progress of SRL.

## 5 Conclusions and Future Work

Verb selection is challenging because verb usage highly depends on the usage context, which is hard to capture and represent. In this paper, we propose to utilize the output of an SRL system to explicitly model verb usage context. We also propose to use the generalized perceptron learning framework to integrate SRL-derived features with other features. Experimental results show that our method outperforms the SMT-based system and achieves state-of-the-art performance when SRL-related features and other local features are integrated. We also show that, for cases where the particular verb usage mainly depends on its distant arguments, a system with only SRL-derived features performs much better than the system with only local features.

In the future, we plan to automatically construct confusion sets, expand our approach to more verbs and test our approach on a larger size of real data. We will try to combine the outputs of several SRL systems to make our system more robust. We also plan to further validate the effectiveness of the SRL-derived features under other learning methods like SVMs.

## Acknowledgment

We thank the anonymous reviewers for their valuable comments. We also thank Changning Huang, Yunbo Cao, Dongdong Zhang, Henry Li and Mu Li for helpful discussions.

## References

Francis Bond, Kentaro Ogura, and Satoru Ikehara. 1994. Countability and number in Japanese to English machine translation. *Proc. of the 15th conference on Computational Linguistics*, pages 32-38.

Chris Brockett, William B. Dolan, and Michael Gamon. 2006. Correcting ESL errors using phrasal SMT techniques. *Proc. of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting on Association for Computational Linguistics*, pages 249-256.

Michael Collins. 2002. Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms. *Proc. of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, pages 1-8.

Jens Eeg-Olofsson and Ola Knutsson. 2003. Automatic Grammar Checking for Second Language Learners – the Use of Prepositions. *Proc. of NoDaliDa*.

Michael Gamon, Jianfeng Gao, Chris Brockett, Alexandre Klementiev, William B. Dolan, Dmitrtiy Belenko, and Lucy Vanderwende. 2008. Using Contextual Speller Techniques and Language Modeling for ESL Error Correction. *Proc. of the International Joint Conference on Natural Language Processing*.

Shichun Gui and Huizhong Yang. 2002. *Chinese Learner English Corpus*. Shanghai Foreign Languages Education Press, Shanghai, China.

Julia E. Heine. 1998. Definiteness predictions for Japanese noun phrases. *Proc. of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, pages 519-525.

John Lee and Stephanie Seneff. 2008. Correcting misuse of verb forms. *Proc. of the 46th Annual Meeting on Association for Computational Linguistics,* pages 174-182.

Ting Liu, Ming Zhou, Jianfeng Gao, Endong Xun and Changning Huang. 2000. PENS: A Machine-aided English Writing System for Chinese Users. *Proc. of the 38th Annual Meeting on Association for Computational Linguistics*, pages 529-536.

Lluís Màrquez. 2009. *Semantic Role Labeling Past, Present and Future*, Tutorial of ACL-IJCNLP 2009.

Dan Moldovan, Adriana Badulescu, Marta Tatu, Daniel Antohe and Roxana Girju. 2004. Models for the semantic classification of noun phrases. *Proc. of the HLT-NAACL Workshop on Computational Lexical Semantics*, pages 60-67.

Srini Narayanan and Sanda Harabagiu. 2004. Question answering based on semantic structures. *Proc. of the 20th International Conference on Computational Linguistics*, pages 693-701.

Franz J. Och and Hermann Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Journal of Computational Linguistics,* 30(4), pages 417-449.

Sebastian Riedel and Ivan Meza-Ruiz. 2008. Collective semantic role labelling with Markov Logic. *Proc. of the Twelfth Conference on Computational Natural Language Learning*, pages 193-197.

Andreas Stolcke. 2002. SRILM -- An Extensible Language Modeling Toolkit. *Proc. of International Conference on Spoken Language Processing*, pages: 901-904.

Mihai Surdeanu, Lluis Màrquez, Xavier Carreras, and Pere R. Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research*, page 105-151.

Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. 2003. Using predicate-argument structures for information extraction. *Proc. of the 41st Annual Meeting on Association for Computational Linguistics*, pages 8-15.

Joel R. Tetreault and Martin Chodorow. 2008. The ups and downs of preposition error detection in ESL writing. *Proc. of the 22nd international Conference on Computational Linguistics*, pages 865-872.

Vladimir N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

Patrick Ye and Timothy Baldwin. 2006. Verb Sense Disambiguation Using Selectional Preferences Extracted with a State-of-the-art Semantic Role Labeler. *Proc. of the Australasian Language Technology Workshop*, pages 141-148.

Xing Yi, Jianfeng Gao, and William B. Dolan. 2008. A Web-based English Proofing System for English as a Second Language Users. *Proc. of International Joint Conference on Natural Language Processing*, pages 619-624.