A Cost Sensitive Part-of-Speech Tagging: Differentiating Serious Errors from Minor Errors

Hyun-Je Song[1] **Jeong-Woo Son**[1] **Tae-Gil Noh**[2] **Seong-Bae Park**[1,3] **Sang-Jo Lee**[1]

[1]School of Computer Sci. & Eng.
Kyungpook Nat'l Univ.
Daegu, Korea

[2]Computational Linguistics
Heidelberg University
Heidelberg, Germany

[3]NLP Lab.
Dept. of Computer Science
University of Illinois at Chicago

{hjsong,jwson,tgnoh}@sejong.knu.ac.kr sbpark@uic.edu sjlee@knu.ac.kr

Abstract

All types of part-of-speech (POS) tagging errors have been equally treated by existing taggers. However, the errors are not equally important, since some errors affect the performance of subsequent natural language processing (NLP) tasks seriously while others do not. This paper aims to minimize these serious errors while retaining the overall performance of POS tagging. Two gradient loss functions are proposed to reflect the different types of errors. They are designed to assign a larger cost to serious errors and a smaller one to minor errors. Through a set of POS tagging experiments, it is shown that the classifier trained with the proposed loss functions reduces serious errors compared to state-of-the-art POS taggers. In addition, the experimental result on text chunking shows that fewer serious errors help to improve the performance of subsequent NLP tasks.

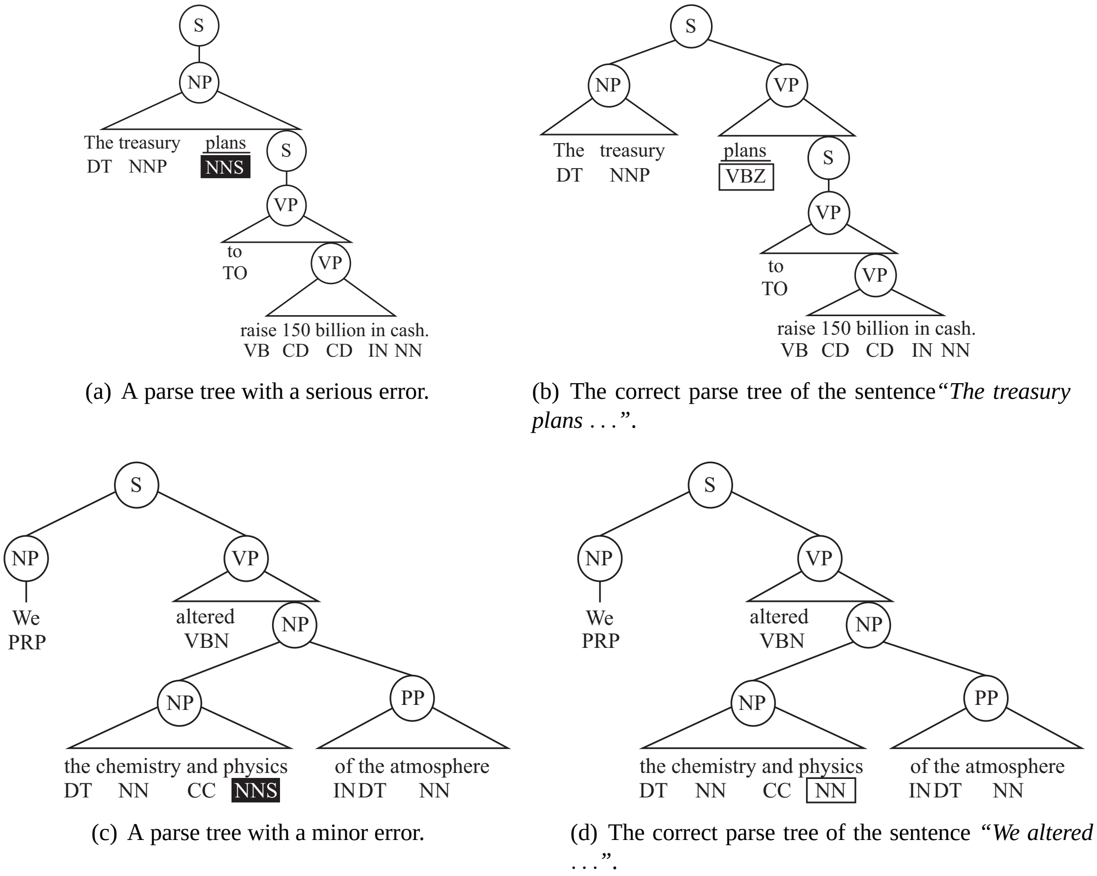
1 Introduction

Part-of-speech (POS) tagging is needed as a preprocessor for various natural language processing (NLP) tasks such as parsing, named entity recognition (NER), and text chunking. Since POS tagging is normally performed in the early step of NLP tasks, the errors in POS tagging are critical in that they affect subsequent steps and often lower the overall performance of NLP tasks.

Previous studies on POS tagging have shown high performance with machine learning techniques (Ratnaparkhi, 1996; Brants, 2000; Lafferty et al., 2001). Among the types of machine learning approaches, supervised machine learning techniques were commonly used in early studies on POS tagging. With the characteristics of a language (Ratnaparkhi, 1996; Kudo et al., 2004) and informative features for POS tagging (Toutanova and Manning, 2000), the state-of-the-art supervised POS tagging achieves over 97% of accuracy (Shen et al., 2007; Manning, 2011). This performance is generally regarded as the maximum performance that can be achieved by supervised machine learning techniques. There have also been many studies on POS tagging with semi-supervised (Subramanya et al., 2010; Søgaard, 2011) or unsupervised machine learning methods (Berg-Kirkpatrick et al., 2010; Das and Petrov, 2011) recently. However, there still exists room to improve supervised POS tagging in terms of error differentiation.

It should be noted that not all errors are equally important in POS tagging. Let us consider the parse trees in Figure 1 as an example. In Figure 1(a), the word "*plans*" is mistagged as a noun where it should be a verb. This error results in a wrong parse tree that is severely different from the correct tree shown in Figure 1(b). The verb phrase of the verb "*plans*" in Figure 1(b) is discarded in Figure 1(a) and the whole sentence is analyzed as a single noun phrase. Figure 1(c) and (d) show another tagging error and its effect. In Figure 1(c), a noun is tagged as a NNS (plural noun) where its correct tag is NN (singular or mass noun). However, the error in Figure 1(c) affects only locally the noun phrase to which "*physics*" belongs. As a result, the general structure of the parse tree in Figure 1(c) is nearly the same as

Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 1025–1034,
Jeju, Republic of Korea, 8-14 July 2012.



(a) A parse tree with a serious error.

(b) The correct parse tree of the sentence*"The treasury plans . . . "*.

(c) A parse tree with a minor error.

(d) The correct parse tree of the sentence *"We altered . . . "*.

Figure 1: An example of POS tagging errors

the correct one in Figure 1(d). That is, a sentence analyzed with this type of error would yield a correct or near-correct result in many NLP tasks such as machine translation and text chunking.

The goal of this paper is to differentiate the serious POS tagging errors from the minor errors. POS tagging is generally regarded as a classification task, and zero-one loss is commonly used in learning classifiers (Altun et al., 2003). Since zero-one loss considers all errors equally, it can not distinguish error types. Therefore, a new loss is required to incorporate different error types into the learning machines.

This paper proposes two gradient loss functions to reflect differences among POS tagging errors. The functions assign relatively small cost to minor errors, while larger cost is given to serious errors. They are applied to learning multiclass support vector machines (Tsochantaridis et al., 2004) which is trained to minimize the serious errors. Overall accuracy of this SVM is not improved against the state-of-the-art POS tagger, but the serious errors are significantly reduced with the proposed method. The effect of the fewer serious errors is shown by applying it to the well-known NLP task of text chunking. Experimental results show that the proposed method achieves a higher F1-score compared to other POS taggers.

The rest of the paper is organized as follows. Section 2 reviews the related studies on POS tagging. In Section 3, serious and minor errors are defined, and it is shown that both errors are observable in a general corpus. Section 4 proposes two new loss functions for discriminating the error types in POS tagging. Experimental results are presented in Section 5. Finally, Section 6 draws some conclusions.

2 Related Work

The POS tagging problem has generally been solved by machine learning methods for sequential label-

Tag category	POS tags
Substantive	NN, NNS, NNP, NNPS, CD, PRP, PRP$
Predicate	VB, VBD, VBG, VBN, VBP, VBZ, MD, JJ, JJR, JJS
Adverbial	RB, RBR, RBS, RP, UH, EX, WP, WP$, WRB, CC, IN, TO
Determiner	DT, PDT, WDT
Etc	FW, SYM, POS, LS

Table 1: Tag categories and POS tags in Penn Tree Bank tag set

ing. In early studies, rich linguistic features and supervised machine learning techniques are applied by using annotated corpora like the Wall Street Journal corpus (Marcus et al., 1994). For instance, Ratnaparkhi (1996) used a maximum entropy model for POS tagging. In this study, the features for rarely appearing words in a corpus are expanded to improve the overall performance. Following this direction, various studies have been proposed to extend informative features for POS tagging (Toutanova and Manning, 2000; Toutanova et al., 2003; Manning, 2011). In addition, various supervised methods such as HMMs and CRFs are widely applied to POS tagging. Lafferty et al. (2001) adopted CRFs to predict POS tags. The methods based on CRFs not only have all the advantages of the maximum entropy markov models but also resolve the well-known problem of label bias. Kudo et al. (2004) modified CRFs for non-segmented languages like Japanese which have the problem of word boundary ambiguity.

As a result of these efforts, the performance of state-of-the-art supervised POS tagging shows over 97% of accuracy (Toutanova et al., 2003; Giménez and Màrquez, 2004; Tsuruoka and Tsujii, 2005; Shen et al., 2007; Manning, 2011). Due to the high accuracy of supervised approaches for POS tagging, it has been deemed that there is no room to improve the performance on POS tagging in supervised manner. Thus, recent studies on POS tagging focus on semi-supervised (Spoustová et al., 2009; Subramanya et al., 2010; Søgaard, 2011) or unsupervised approaches (Haghighi and Klein, 2006; Goldwater and Griffiths, 2007; Johnson, 2007; Graca et al., 2009; Berg-Kirkpatrick et al., 2010; Das and Petrov, 2011). Most previous studies on POS tagging have focused on how to extract more linguistic features or how to adopt supervised or unsupervised approaches based on a single evaluation measure, *accuracy*. However, with a different viewpoint for errors on POS tagging, there is still some room to improve the performance of POS tagging for subsequent NLP tasks, even though the overall accuracy can not be much improved.

In ordinary studies on POS tagging, costs of errors are equally assigned. However, with respect to the performance of NLP tasks relying on the result of POS tagging, errors should be treated differently. In the machine learning community, cost sensitive learning has been studied to differentiate costs among errors. By adopting different misclassification costs for each type of errors, a classifier is optimized to achieve the lowest expected cost (Elkan, 2001; Cai and Hofmann, 2004; Zhou and Liu, 2006).

3 Error Analysis of Existing POS Tagger

The effects of POS tagging errors to subsequent NLP tasks vary according to their type. Some errors are serious, while others are not. In this paper, the seriousness of tagging errors is determined by categorical structures of POS tags. Table 1 shows the Penn tree bank POS tags and their categories. There are five categories in this table: *substantive*, *predicate*, *adverbial*, *determiner*, and *etc*. Serious tagging errors are defined as misclassifications among the categories, while minor errors are defined as misclassifications within a category. This definition follows the fact that POS tags in the same category form similar syntax structures in a sentence (Zhao and Marcus, 2009). That is, inter-category errors are treated as serious errors, while intra-category errors are treated as minor errors.

Table 2 shows the distribution of inter-category and intra-category errors observed in section 22–24 of the WSJ corpus (Marcus et al., 1994) that is tagged by the Stanford Log-linear Part-Of-Speech

		Predicted category				
		Substantive	Predicate	Adverbial	Determiner	Etc
True category	Substantive	614	**479**	**32**	**10**	**15**
	Predicate	**585**	743	**107**	**2**	**14**
	Adverbial	**41**	**156**	500	**42**	**2**
	Determiner	**13**	**7**	**47**	24	**0**
	Etc	**23**	**11**	**3**	**1**	0

Table 2: The distribution of tagging errors on WSJ corpus by Stanford Part-Of-Speech Tagger.

Tagger (Manning, 2011) (trained with WSJ sections 00–18). In this table, bold numbers denote inter-category errors while all other numbers show intra-category errors. The number of total errors is 3,471 out of 129,654 words. Among them, 1,881 errors (54.19%) are intra-category, while 1,590 of the errors (45.81%) are inter-category. If we can reduce these inter-category errors under the cost of minimally increasing intra-category errors, the tagging results would improve in quality.

Generally in POS tagging, all tagging errors are regarded equally in importance. However, inter-category and intra-category errors should be distinguished. Since a machine learning method is optimized by a loss function, inter-category errors can be efficiently reduced if a loss function is designed to handle both types of errors with different cost. We propose two loss functions for POS tagging and they are applied to multiclass Support Vector Machines.

4 Learning SVMs with Class Similarity

POS tagging has been solved as a sequential labeling problem which assumes dependency among words. However, by adopting sequential features such as POS tags of previous words, the dependency can be partially resolved. If it is assumed that words are independent of one another, POS tagging can be regarded as a multiclass classification problem. One of the best solutions for this problem is by using an SVM.

4.1 Training SVMs with Loss Function

Assume that a training data set $D = \{(x_1, y_1), (x_2, y_2), \ \ldots, (x_l, y_l)\}$ is given where $x_i \in \mathbf{R}^d$ is an instance vector and $y_i \in \{+1, -1\}$ is its class label. SVM finds an optimal hyperplane satisfying

$$\begin{aligned} x_i \cdot w + b \geq +1 \quad &\text{for} \quad y_i = +1, \\ x_i \cdot w + b \leq -1 \quad &\text{for} \quad y_i = -1, \end{aligned}$$

where w and b are parameters to be estimated from training data D. To estimate the parameters, SVMs minimizes a hinge loss defined as

$$\begin{aligned} \xi_i &= L_{hinge}(y_i, w \cdot x_i + b) \\ &= \max\{0, 1 - y_i \cdot (w \cdot x_i + b)\}. \end{aligned}$$

With regularizer $||w||^2$ to control model complexity, the optimization problem of SVMs is defined as

$$\min_{w,\xi} \frac{1}{2} ||w||^2 + C \sum_{i=1}^{l} \xi_i,$$

subject to

$$y_i(x_i \cdot w + b) \geq 1 - \xi_i, \text{ and } \xi_i \geq 0 \ \ \forall i,$$

where C is a user parameter to penalize errors.

Crammer et al. (2002) expanded the binary-class SVM for multiclass classifications. In multiclass SVMs, by considering all classes the optimization of SVM is generalized as

$$\min_{w,\xi} \frac{1}{2} \sum_{k \in K} ||w_k||^2 + C \sum_{i=1}^{l} \xi_i,$$

with constraints

$$\begin{aligned} (w_{y_i} \cdot \phi(x_i, y_i)) - (w_k \cdot \phi(x_i, k)) \geq 1 - \xi_i, \\ \xi_i \geq 0 \ \ \forall i, \ \ \forall k \in K \setminus y_i, \end{aligned}$$

where $\phi(x_i, y_i)$ is a combined feature representation of x_i and y_i, and K is the set of classes.

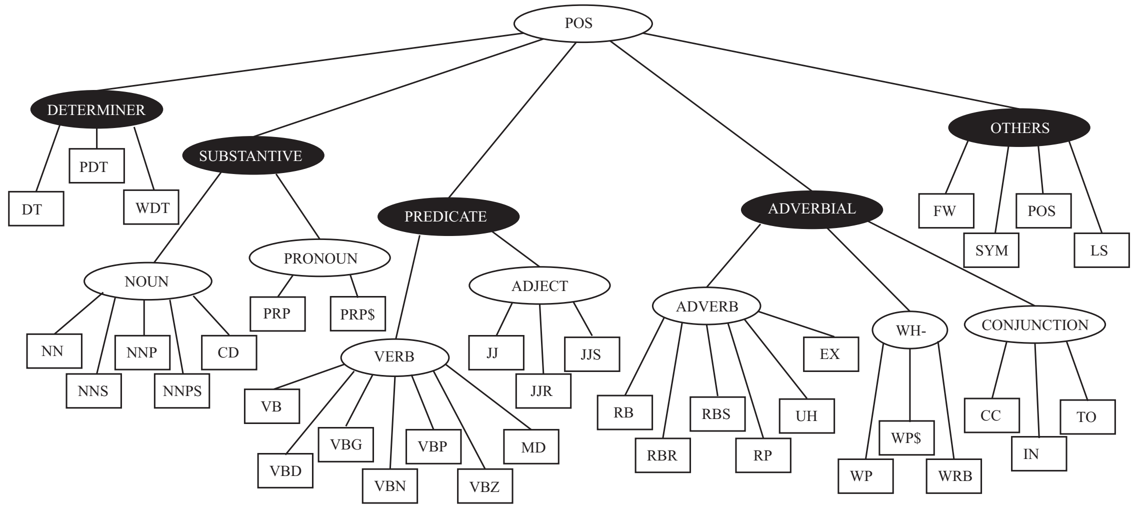


Figure 2: A tree structure of POS tags.

Since both binary and multiclass SVMs adopt a hinge loss, the errors between classes have the same cost. To assign different cost to different errors, Tsochantaridis et al. (2004) proposed an efficient way to adopt arbitrary loss function, $L(y_i, y_j)$ which returns zero if $y_i = y_j$, otherwise $L(y_i, y_j) > 0$. Then, the hinge loss ξ_i is re-scaled with the inverse of the additional loss between two classes. By scaling slack variables with the inverse loss, margin violation with high loss $L(y_i, y_j)$ is more severely restricted than that with low loss. Thus, the optimization problem with $L(y_i, y_j)$ is given as

$$\min_{w,\xi} \frac{1}{2} \sum_{k \in K} ||w_k||^2 + C \sum_{i=1}^{l} \xi_i, \qquad (1)$$

with constraints

$$(w_{y_i} \cdot \phi(x_i, y_i)) - (w_k \cdot \phi(x_i, k)) \geq 1 - \frac{\xi_i}{L(y_i, k)},$$
$$\xi_i \geq 0 \ \ \forall i, \ \ \forall k \in K \setminus y_i,$$

With the Lagrange multiplier α, the optimization problem in Equation (1) is easily converted to the following dual quadratic problem.

$$\min_{\alpha} \frac{1}{2} \sum_{i,j}^{l} \sum_{k_i \in K \setminus y_i} \sum_{k_j \in K \setminus y_j} \alpha_{i,k_i} \alpha_{j,k_j} \times$$
$$J(x_i, y_i, k_i) J(x_j, y_j, k_j) - \sum_{i}^{l} \sum_{k_i \in K \setminus y_i} \alpha_{i,k_i},$$

with constraints

$$\alpha \geq 0 \text{ and } \sum_{k_i \in K \setminus y_i} \frac{\alpha_{i,k_i}}{L(y_i, k_i)} \leq C, \ \ \forall i = 1, \cdots, l,$$

where $J(x_i, y_i, k_i)$ is defined as

$$J(x_i, y_i, k_i) = \phi(x_i, y_i) - \phi(x_i, k_i).$$

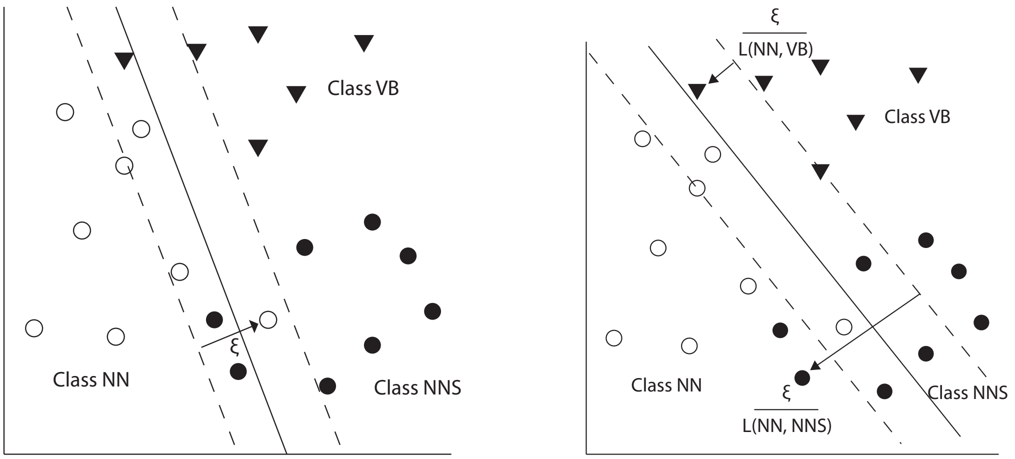
4.2 Loss Functions for POS tagging

To design a loss function for POS tagging, this paper adopts categorical structures of POS tags. The simplest way to reflect the structure of POS tags shown in Table 1 is to assign larger cost to inter-category errors than to intra-category errors. Thus, the loss function with the categorical structure in Table 1 is defined as

$$L_c(y_i, y_j) = \begin{cases} 0 & \text{if } y_i = y_j, \\ \delta & \text{if } y_i \neq y_j \text{ but they belong} \\ & \text{to the same POS category,} \\ 1 & \text{otherwise,} \end{cases} \qquad (2)$$

where $0 < \delta < 1$ is a constant to reduce the value of $L_c(y_i, y_j)$ when y_i and y_j are similar. As shown in this equation, inter-category errors have larger cost than intra-category errors. This loss $L_c(y_i, y_j)$ is named as *category loss*.

The loss function $L_c(y_i, y_j)$ is designed to reflect the categories in Table 1. However, the structure of POS tags can be represented as a more complex structure. Let us consider the category, **predicate**.



(a) Multiclass SVMs with hinge loss

(b) Multiclass SVMs with the proposed loss function

Figure 3: Effect of the proposed loss function in multiclass SVMs

This category has ten POS tags, and can be further categorized into two sub-categories: **verb** and **adject**. Figure 2 represents a categorical structure of POS tags as a tree with five categories of POS tags and their seven sub-categories.

To express the tree structure of Figure 2 as a loss, another loss function $L_t(y_i, y_j)$ is defined as

$$L_t(y_i, y_j) = \frac{1}{2}[Dist(P_{i,j}, y_i) + Dist(P_{i,j}, y_j)] \times \gamma, \qquad (3)$$

where $P_{i,j}$ denotes the nearest common parent of both y_i and y_j, and the function $Dist(P_{i,j}, y_i)$ returns the number of steps from $P_{i,j}$ to y_i. The user parameter γ is a scaling factor of a unit loss for a single step. This loss $L_t(y_i, y_j)$ returns large value if the distance between y_i and y_j is far in the tree structure, and it is named as *tree loss*.

As shown in Equation (1), two proposed loss functions adjust margin violation between classes. They basically assign less value for intra-category errors than inter-category errors. Thus, a classifier is optimized to strictly keep inter-category errors within a smaller boundary. Figure 3 shows a simple example. In this figure, there are three POS tags and two categories. NN (singular or mass noun) and NNS (plural noun) belong to the same category, while VB (verb, base form) is in another category. Figure 3(a) shows the decision boundary of NN based on hinge loss. As shown in this figure, a single ξ is applied for the margin violation among all classes. Figure 3(b) also presents the decision boundary of NN, but it is determined with the proposed loss function. In this figure, the margin violation is applied differently to inter-category (NN to VB) and intra-category (NN to NNS) errors. It results in reducing errors between NN and VB even if the errors between NN and NNS could be slightly increased.

5 Experiments

5.1 Experimental Setting

Experiments are performed with a well-known standard data set, the Wall Street Journal (WSJ) corpus. The data is divided into training, development and test sets as in (Toutanova et al., 2003; Tsuruoka and Tsujii, 2005; Shen et al., 2007). Table 3 shows some simple statistics of these data sets. As shown in this table, training data contains 38,219 sentences with 912,344 words. In the development data set, there are 5,527 sentences with about 131,768 words, those in the test set are 5,462 sentences and 129,654 words. The development data set is used only to select δ in Equation (2) and γ in Equation (3).

Table 4 shows the feature set for our experiments. In this table, w_i and t_i denote the lexicon and POS tag for the i-th word in a sentence respectively. We use almost the same feature set as used in (Tsuruoka and Tsujii, 2005) including word features, tag fea-

	Training	Develop	Test
Section	0–18	19–21	22–24
# of sentences	38,219	5,527	5,462
# of terms	912,344	131,768	129,654

Table 3: Simple statistics of experimental data

Feature Name	Description
Word features	w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2} $w_{i-1} \cdot w_i$, $w_i \cdot w_{i+1}$
Tag features	t_{i-2}, t_{i-1}, t_{i+1}, t_{i+2} $t_{i-2} \cdot t_{i-1}$, $t_{i+1} \cdot t_{i+2}$ $t_{i-2} \cdot t_{i-1} \cdot t_{i+1}$, $t_{i-1} \cdot t_{i+1} \cdot t_{i+2}$ $t_{i-2} \cdot t_{i-1} \cdot t_{i+1} \cdot t_{i+2}$
Tag/Word combination	$t_{i-2} \cdot w_i$, $t_{i-1} \cdot w_i$, $t_{i+1} \cdot w_i$, $t_{i+2} \cdot w_i$ $t_{i-1} \cdot t_{i+1} \cdot w_i$
Prefix features	prefixes of w_i (up to length 9)
Suffix features	suffixes of w_i (up to length 9)
Lexical features	whether w_i contains capitals whether w_i has a number whether w_i has a hyphen whether w_i is all capital whether w_i starts with capital and locates at the middle of sentence

Table 4: Feature template for experiments

	Error (%)	# of Intra error	# of Inter error
(Giménez and Màrquez, 2004)	2.84	1,995 (54.11%)	1,692 (45.89%)
(Tsuruoka and Tsujii, 2005)	2.85	-	-
(Shen et al., 2007)	**2.67**	**1,856 (53.52%)**	1,612 (46.48%)
(Manning, 2011)	2.68	1,881 (54.19%)	1,590 (45.81%)
CL-MSVM ($\delta = 0.4$)	2.69	1,916 (55.01%)	**1,567 (44.99%)**
TL-MSVM ($\gamma = 0.6$)	2.68	1,904 (54.74%)	**1,574 (45.26%)**

Table 5: Comparison with the previous works

tures, word/tag combination features, prefix and suffix features as well as lexical features. The POS tags for words are obtained from a two-pass approach proposed by Nakagawa et al. (2001).

In the experiments, two multiclass SVMs with the proposed loss functions are used. One is CL-MSVM with category loss and the other is TL-MSVM with tree loss. A linear kernel is used for both SVMs.

5.2 Experimental Results

CL-MSVM with $\delta = 0.4$ shows the best overall performance on the development data where its error rate is as low as 2.71%. $\delta = 0.4$ implies that the cost of intra-category errors is set to 40% of that of inter-category errors. The error rate of TL-MSVM is 2.69% when γ is 0.6. δ = 0.4 and γ = 0.6 are set in the all experiments below.

Table 5 gives the comparison with the previous work and proposed methods on the test data. As can be seen from this table, the best performing algorithms achieve near 2.67% error rate (Shen et al., 2007; Manning, 2011). CL-MSVM and TL-MSVM achieve an error rate of 2.69% and 2.68% respectively. Although overall error rates of CL-MSVM and TL-MSVM are not improved compared to the previous state-of-the-art methods, they show reasonable performance.

For inter-category error, CL-MSVM achieves the best performance. The number of inter-category error is 1,567, which shows 23 errors reduction compared to previous best inter-category result by (Manning, 2011). TL-MSVM also makes 16 less inter-category errors than Manning's tagger. When compared with Shen's tagger, both CL-MSVM and TL-MSVM make far less inter-category errors even if their overall performance is slightly lower than that of Shen's tagger. However, the intra-category error rate of the proposed methods has some slight increases. The purpose of proposed methods is to minimize inter-category errors but preserving overall performance. From these results, it can be found that the proposed methods which are trained with the proposed loss functions do differentiate serious and minor POS tagging errors.

5.3 Chunking Experiments

The task of chunking is to identify the non-recursive cores for various types of phrases. In chunking, the POS information is one of the most crucial aspects in identifying chunks. Especially inter-category POS errors seriously affect the performance of chunking because they are more likely to mislead the chunk compared to intra-category errors.

Here, chunking experiments are performed with

POS tagger	Accuracy (%)	Precision	Recall	F1-score
(Shen et al., 2007)	96.08	94.03	93.75	93.89
(Manning, 2011)	96.08	94	93.8	93.9
CL-MSVM ($\delta = 0.4$)	**96.13**	**94.1**	**93.9**	**94.00**
TL-MSVM ($\gamma = 0.6$)	96.12	**94.1**	**93.9**	**94.00**

Table 6: The experimental results for chunking

a data set provided for the CoNLL-2000 shared task. The training data contains 8,936 sentences with 211,727 words obtained from sections 15–18 of the WSJ. The test data consists of 2,012 sentences and 47,377 words in section 20 of the WSJ. In order to represent chunks, an IOB model is used, where every word is tagged with a chunk label extended with B (the beginning of a chunk), I (inside a chunk), and O (outside a chunk). First, the POS information in test data are replaced to the result of our POS tagger. Then it is evaluated using trained chunking model. Since CRFs (Conditional Random Fields) has been shown near state-of-the-art performance in text chunking (Fei Sha and Fernando Pereira, 2003; Sun et al., 2008), we use CRF++, an open source CRF implementation by Kudo (2005), with default feature template and parameter settings of the package. For simplicity in the experiments, the values of δ in Equation (2) and γ in Equation (3) are set to be 0.4 and 0.6 respectively which are same as the previous section.

Table 6 gives the experimental results of text chunking according to the kinds of POS taggers including two previous works, CL-MSVM, and TL-MSVM. Shen's tagger and Manning's tagger show nearly the same performance. They achieve an accuracy of 96.08% and around 93.9 F1-score. On the other hand, CL-MSVM achieves 96.13% accuracy and 94.00 F1-score. The accuracy and F1-score of TL-MSVM are 96.12% and 94.00. Both CL-MSVM and TL-MSVM show slightly better performances than other POS taggers. As shown in Table 5, both CL-MSVM and TL-MSVM achieve lower accuracies than other methods, while their inter-category errors are less than that of other experimental methods. Thus, the improvement of CL-MSVM and TL-MSVM implies that, for the subsequent natural language processing, a POS tagger should considers different cost of tagging errors.

6 Conclusion

In this paper, we have shown that supervised POS tagging can be improved by discriminating inter-category errors from intra-category ones. An inter-category error occurs by mislabeling a word with a totally different tag, while an intra-category error is caused by a similar POS tag. Therefore, inter-category errors affect the performances of subsequent NLP tasks far more than intra-category errors. This implies that different costs should be considered in training POS tagger according to error types.

As a solution to this problem, we have proposed two gradient loss functions which reflect different costs for two error types. The cost of an error type is set according to (i) categorical difference or (ii) distance in the tree structure of POS tags. Our POS experiment has shown that if these loss functions are applied to multiclass SVMs, they could significantly reduce inter-category errors. Through the text chunking experiment, it is shown that the multiclass SVMs trained with the proposed loss functions which generate fewer inter-category errors achieve higher performance than existing POS taggers.

We have shown that cost sensitive learning can be applied to POS tagging only with multiclass SVMs. However, the proposed loss functions are general enough to be applied to other existing POS taggers. Most supervised machine learning techniques are optimized on their loss functions. Therefore, the performance of POS taggers based on supervised machine learning techniques can be improved by applying the proposed loss functions to learn their classifiers.

Acknowledgments

This research was supported by the Converging Research Center Program funded by the Ministry of Education, Science and Technology (2011K000659).

References

Yasemin Altun, Mark Johnson, and Thomas Hofmann. 2003. Investigating Loss Functions and Optimization Methods for Discriminative Learning of Label Sequences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 145–152.

Talyor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless Unsupervised Learning with Features. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*. pp. 582–590.

Thorsten Brants. 2000. TnT-A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference*. pp. 224–231.

Lijuan Cai and Thomas Hofmann. 2004. Hierarchical Document Categorization with Support Vector Machines. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*. pp. 78–87.

Koby Crammer, Yoram Singer. 2002. On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines. *Journal of Machine Learning Research*, Vol. 2. pp. 265–292.

Dipanjan Das and Slav Petrov. 2011. Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections. In *Proceedings of the 49th Annual Meeting of the Association of Computational Linguistics*. pp. 600–609.

Charles Elkan. 2001. The Foundations of Cost-Sensitive Learning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*. pp. 973–978.

Jesús Giménez and Lluís Màrquez. 2004. SVMTool: A general POS tagger generator based on Support Vector Machines. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation*. pp. 43–46.

Sharon Goldwater and Thomas T. Griffiths. 2007. A fully Bayesian Approach to Unsupervised Part-of-Speech Tagging. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. pp. 744–751.

Joao Graca, Kuzman Ganchev, Ben Taskar, and Fernando Pereira. 2009. Posterior vs Parameter Sparsity in Latent Variable Models. In *Advances in Neural Information Processing Systems 22*. pp. 664–672.

Aria Haghighi and Dan Klein. 2006. Prototype-driven Learning for Sequence Models. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*. pp. 320–327.

Mark Johnson. 2007. Why doesn't EM find good HMM POS-taggers? In *Proceedings of the 2007 Joint Meeting of the Conference on Empirical Methods in Natural Language Processing and the Conference on Computational Natural Language Learning*. pp. 296–305.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 230–237.

Taku Kudo. 2005. CRF++: Yet another CRF toolkit. http://crfpp.sourceforge.net.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*. pp. 282–289.

Christopher D. Manning. 2011. Part-of-Speech Tagging from 97% to 100%: Is It Time for Some Linguistics?. In *Proceedings of the 12th International Conference on Intelligent Text Processing and Computational Linguistics*. pp. 171–189.

Tetsuji Nakagawa, Taku Kudo, and Yuji Matsumoto. 2001. Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines. In *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium*. pp. 325–331.

Adwait Ratnaparkhi. 1996. A Maximum Entropy Model for Part-Of-Speech Tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 133–142.

Fei Sha and Fernando Pereira. 2003. Shallow Parsing with Conditional Random Fields. In *Proceedings of the Human Language Technology and North American Chapter of the Association for Computational Linguistics*. pp. 213–220.

Libin Shen, Giorgio Satta, and Aravind K. Joshi 2007. Guided Learning for Bidirectional Sequence Classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. pp. 760–767.

Anders Søgaard 2011. Semisupervised condensed nearest neighbor for part-of-speech tagging. In *Proceedings of the 49th Annual Meeting of the Association of Computational Linguistics*. pp. 48–52.

Drahomíra "johanka" Spoustovà, Jan Hajič, Jan Raab, and Miroslav Spousta 2009. Semi-supervised training for the averaged perceptron POS tagger. In *Proceedings of the European Chapter of the Association for Computational Linguistics*. pp. 763–771.

Amarnag Subramanya, Slav Petrov and Fernando Pereira 2010. Efficient Graph-Based Semi-Supervised Learning of Structured Tagging Models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 167–176.

Xu Sun, Louis-Philippe Morency, Daisuke Okanohara and Jun'ichi Tsujii 2008. Modeling Latent-Dynamic in Shallow Parsing: A Latent Conditional Model with Improved Inference. In *Proceedings of the 22nd International Conference on Computational Linguistics*. pp. 841–848.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network.

In *Proceedings of the Human Language Technology and North American Chapter of the Association for Computational Linguistics*. pp. 252–259.

Kristina Toutanova and Christopher D. Manning. 2000. Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 63–70.

Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemi Altun. 2004. Support Vector Learning for Interdependent and Structured Output Spaces. In *Proceedings of the 21st International Conference on Machine Learning*. pp. 104–111.

Yoshimasa Tsuruoka and Jun'ichi Tsujii. 2005. Bidirectional Inference with the Easiest-First Strategy for Tagging Sequence Data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 467–474.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1994. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, Vol. 19, No.2 . pp. 313–330.

Qiuye Zhao and Mitch Marcus. 2009. A Simple Unsupervised Learner for POS Disambiguation Rules Given Only a Minimal Lexicon. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. pp. 688–697.

Zhi-Hua Zhou and Xu-Ying Liu 2006. On Multi-Class Cost-Sensitive Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. pp. 567–572.