# A Modified Fixed-Threshold SMO for 1-Slack Structural SVMs

Changki Lee and Myung-Gil Jang

**In this paper, we describe a modified fixed-threshold sequential minimal optimization (FSMO) for 1-slack structural support vector machine (SVM) problems. Because the modified FSMO uses the fact that the formulation of 1-slack structural SVMs has no bias, it breaks down the quadratic programming (QP) problems of 1-slack structural SVMs into a series of smallest QP problems, each involving only one variable. For various test sets, the modified FSMO is as accurate as existing structural SVM implementations (*n*-slack and 1-slack SVM-struct) but is faster on large data sets.**

**Keywords: 1-slack structural SVM, fixed-threshold sequential minimal optimization, modified FSMO.**

## I. Introduction

Large-margin methods for structured output prediction, such as maximum-margin Markov networks [1] and structural support vector machines (SVMs) [2], have recently received substantial interest in natural language processing [3], bioinformatics [4], and information retrieval [5].

For structural SVMs, Tsochantaridis presented a cutting-plane algorithm that takes $O(1/\varepsilon^2)$ iterations to reach a desired precision $\varepsilon$ [2]. For training of a structural SVM, Tsochantaridis used a standard SVM solver, namely, SVM-light, for solving the dual form of a structural SVM (SVM-struct), despite the fact that a structural SVM has no bias. That is, the bias $b$ is fixed at zero [2]. Sequential minimal optimization (SMO) [6] has also been applied to large-margin methods [1], [3], [7]. Collins provided an exponentiated gradient method to the structured output problem [8]. Subgradient methods have also been proposed to solve the optimization problem in maximum-margin structured prediction [9]. While not yet explored for structured prediction, the Pegasos algorithm has shown promising performance for binary classification SVMs [10]. Lee and Jang proposed a fixed-threshold sequential minimal optimization (FSMO) algorithm for a structural SVM, and showed that the algorithm is much faster than SVM-struct and LIBSVM (Library for Support Vector Machines) [11].

Recently, Teo suggested a bundle method [12], and Joachims proposed a 1-slack formulation of structural SVMs which is very close to the bundle method [13]. These methods can be viewed as extensions of the method given by Joachims [14] for binary linear SVMs. The 1-slack algorithm is substantially faster than existing methods such as SMO and SVM-light. The convergence rate of the 1-slack algorithm is $O(1/\varepsilon)$.

In this paper, we describe a modified FSMO algorithm for

Manuscript received July 29, 2009; revised Oct. 14, 2009; accepted Oct. 28, 2009.

Changki Lee (phone: +82 42 860 6879, email: leeck@etri.re.kr) and Myung-Gil Jang (email: mgjang@etri.re.kr) are with the Software Research Laboratory, ETRI, Daejeon, Rep. of Korea.

doi:10.4218/etrij.10.0109.0425

  

1-slack structural SVMs. The modified FSMO uses the fact that the formulation of 1-slack structural SVMs has no bias and no linear equality constraint of binary classification SVMs. Therefore, the modified FSMO breaks the quadratic programming (QP) of a structural SVM into a series of smallest QPs, each involving only one variable. By involving only one variable, the modified FSMO is advantageous in that each QP subproblem does not require a working set selection when support vectors are unbounded.

Our main contributions are the following:

- We introduce a modified FSMO algorithm for 1-slack structural SVMs. We show that for the 1-slack structural SVMs, replacing SVM-light with the modified FSMO algorithm shows much faster training results.
- We provide a convergence proof of the modified FSMO algorithm for 1-slack structural SVMs.

For comparison, we extend the Pegasos algorithm for structured prediction.

The rest of this paper is organized as follows. Section II describes the 1-slack structural SVM. Section III describes our proposed modified FSMO algorithm for the 1-slack structural SVM. In section IV, we give an overview of related work. Section V provides the experimental setup and results. The final section gives some concluding remarks.

## II. 1-Slack Structural SVM

Structured classification is the problem of predicting $y$ from $x$ in the case where $y$ has a meaningful internal structure. For example, $x$ might be a word string and $y$ might be a sequence of part of speech labels, or $y$ might be a parse tree of $x$. The approach is to learn the discriminant function $f : X \times Y \to R$ over <*input, output*> pairs from which we can derive a prediction by maximizing $f$ over the response variable for a specific given input $x$. Throughout this paper, we assume $f$ to be linear in some combined feature representation of inputs and outputs $\Psi(x, y)$, $f(x, y; \mathbf{w}) = \mathbf{w}^T \Psi(x, y)$.

The specific form of $\Psi(x, y)$ depends on the nature of the problem. An example of part of speech tagging is shown in Fig. 1.

To deal with problems in which $|Y|$ is very large, Tsochantaridis proposed two approaches, namely, slack rescaling and margin rescaling [2]. In the case of margin rescaling, which we consider in this paper, training a structural SVM amounts to solving the following quadratic program. For convenience, we define $\delta\Psi_i(x_i, y) \equiv \Psi(x_i, y_i) - \Psi(x_i, y)$, where $(x_i, y_i)$ is the training data:

$$\min_{w,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{C}{n}\sum_{i}^{n}\xi_i \quad \text{, s.t.} \quad \forall i, \xi_i \geq 0,$$
$$\forall i, \forall y \in Y \setminus y_i : \mathbf{w}^T \delta\Psi_i(x_i, y) \geq \Delta(y_i, y) - \xi_i. \qquad (1)$$

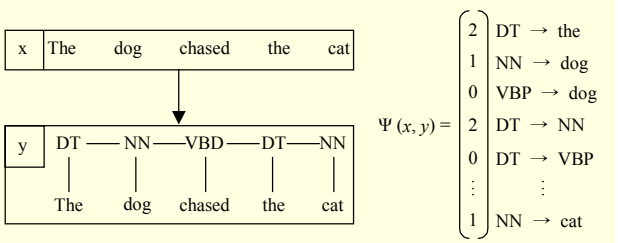


Fig. 1. Example of part of speech tagging model.

This formulation is referred to as the "$n$-slack" structural SVM, since it assigns a different slack variable to each of the $n$ training examples. Tsochantaridis presented a cutting-plane algorithm that requires $O(n/\varepsilon^2)$ constraints for any desired precision $\varepsilon$ [2].

Joachims proposed an alternative formulation of the SVM optimization problem to predict structured outputs [13]. The key idea is to replace the $n$ cutting-plane models of the hinge loss with a single cutting plane model for the sum of the hinge losses. Since there is only a single-slack variable, the new formulation is referred to as "1-slack" structural SVMs.

$$\min_{w,\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\xi \quad \text{, s.t.} \quad \forall i, \xi \geq 0,$$
$$\forall (\hat{y}_1, \cdots, \hat{y}_n) \in Y^n : \frac{1}{n}\mathbf{w}^T \sum_{i=1}^{n} \delta\Psi_i(x_i, \hat{y}_i) \geq \frac{1}{n}\sum_{i=1}^{n}\Delta(y_i, \hat{y}_i) - \xi. \qquad (2)$$

While 1-slack formulations have $|Y|^n$ constraints, one for each possible combination of labels $(\hat{y}_1, \cdots, \hat{y}_n) \in Y^n$, they have only one slack variable $\xi$ that is shared across all constraints. Interestingly, the objective functions of the $n$-slack and 1-slack formulations are equal [13].

Joachims showed that the dual form of the 1-slack formulation has a solution that is extremely sparse with the number of non-zero dual variables independent of the number of training examples and that the convergence rate is $O(1/\varepsilon)$ [13]. To find this solution, Joachims proposed 1-slack cutting plane algorithms. The pseudocode of the algorithm is given in algorithm 1. The algorithm iteratively constructs a working set $S$ of constraints. In each iteration, the algorithm computes the solution over the current $S$ by using SVM-light (line 4), finds the "most violated" constraint (lines 5 to 7), and adds it to the working set $S$ (line 8). The algorithm stops once no constraint can be found that is violated by more than the desired precision $\varepsilon$ (line 9).

## III. Modified FSMO for 1-Slack Structural SVM

In this section, we describe the modified FSMO algorithm for solving the 1-slack structural SVM. Instead of SVM-light,

**Algorithm 1**. 1-slack cutting plane algorithm [13]

1: Input: $(x_1,y_1),\cdots,(x_n,y_n)$, $C$, $\varepsilon$

2: $S \leftarrow \emptyset$

3: repeat

$(\mathbf{w},\xi) \leftarrow \arg\min_{\mathbf{w},\xi>0} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\xi$

4: s.t. $\forall(\hat{y}_1,\cdots,\hat{y}_n) \in S$ :

$$\frac{1}{n}\mathbf{w}^T\sum_{i=1}^{n}\delta\Psi_i(x_i,\hat{y}_i) \geq \frac{1}{n}\mathbf{w}^T\sum_{i=1}^{n}\Delta(y_i,\hat{y}_i) - \xi$$

5: for $i$=1,$\cdots$,$n$ do

6: $\hat{y}_i \leftarrow \max_{\hat{y}\in Y}\{\Delta(y_i,\hat{y}_i) + \mathbf{w}^T\Psi(x_i,\hat{y})\}$

7: end for

8: $S \leftarrow S \cup \{(\hat{y}_1,\cdots,\hat{y}_n)\}$

9: until $\frac{1}{n}\mathbf{w}^T\sum_{i=1}^{n}\Delta(y_i,\hat{y}_i) - \frac{1}{n}\mathbf{w}^T\sum_{i=1}^{n}\delta\Psi_i(x_i,\hat{y}_i) \leq \xi + \varepsilon$

10: return ($\mathbf{w}$,$\xi$)

the modified FSMO is used to solve the dual problem of the 1-slack structural SVM in the 1-slack cutting plane algorithm (line 4 in algorithm 1).

We denote the vectors as follows:

$$\hat{\mathbf{y}} = (\hat{y}_1,\cdots,\hat{y}_n) \in Y^n,$$
$$\Delta(\hat{\mathbf{y}}) = \frac{1}{n}\sum_{i=1}^{n}\Delta(y_i,\hat{y}_i),$$
$$\delta\Psi(\hat{\mathbf{y}}) = \frac{1}{n}\sum_{i=1}^{n}\delta\Psi_i(x_i,\hat{y}_i).$$

We can solve the optimization problem of 1-slack structural SVMs in (2) using standard Lagrangian duality techniques:

$$\min_{w,\xi} L(\mathbf{w},\xi) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\xi + \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}}(\Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}}) - \xi) - \beta\xi,$$
$$\text{s.t. } \forall\hat{\mathbf{y}}, \alpha_{\hat{\mathbf{y}}} \geq 0, \quad \beta \geq 0, \quad \xi \geq 0. \tag{3}$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}}\delta\Psi(\hat{\mathbf{y}}) = 0\,, \tag{4}$$

$$\frac{\partial L}{\partial \xi} = C - \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} - \beta = 0. \tag{5}$$

By substituting (4) and (5) into (3), we obtain the following dual form, which is a QP problem where the objective function $D$ is solely dependent on a set of Lagrangian multipliers:

$$\max_{\alpha} : D(\alpha) = \sum_{\hat{\mathbf{y}}\in Y^n}\Delta(\hat{\mathbf{y}})\alpha_{\hat{\mathbf{y}}} - \frac{1}{2}\sum_{\hat{\mathbf{y}}\in Y^n}\sum_{\hat{\mathbf{y}}'\in Y^n}\alpha_{\hat{\mathbf{y}}}\alpha_{\hat{\mathbf{y}}'}\delta\Psi(\hat{\mathbf{y}})^T\delta\Psi(\hat{\mathbf{y}}'),$$
$$\text{s.t. } \forall\hat{\mathbf{y}}, \quad 0 \leq \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} \leq C, \quad \alpha_{\hat{\mathbf{y}}} \geq 0. \tag{6}$$

The extremum of the object function $D$ is at

$$\frac{\partial D(\alpha)}{\partial \alpha_{\hat{\mathbf{y}}}} = \Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}}) = 0, \tag{7}$$
$$\text{where } \mathbf{w} = \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}}\delta\Psi(\hat{\mathbf{y}}).$$

Let $\alpha_{\hat{\mathbf{y}}}^{\text{new}} = \alpha_{\hat{\mathbf{y}}} + s$ and $\mathbf{w}^{\text{new}} = \mathbf{w} + s\delta\Psi(\hat{\mathbf{y}})$. We can then obtain the following equations from (7):

$$\mathbf{w}^{\text{new T}}\delta\Psi(\hat{\mathbf{y}}) = \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}}) + s\left\|\delta\Psi(\hat{\mathbf{y}})\right\|^2 = \Delta(\hat{\mathbf{y}}), \tag{8}$$

$$s = \frac{\Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}})}{\left\|\delta\Psi(\hat{\mathbf{y}})\right\|^2}. \tag{9}$$

After $s$ is computed, it is changed to satisfy a standard box constraint, $0 \leq \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} \leq C$ and $\alpha_{\hat{\mathbf{y}}} \geq 0$.

$$s^{\text{clipped}} = \max(-\alpha_{\hat{\mathbf{y}}}, \min(C - \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}}, s)). \tag{10}$$

Because 1-slack structural SVMs have no bias, that is, the bias $b$ is fixed at zero, in (2) and (3), (6) has no linear equality constraint of binary classification SVMs. Therefore, the modified FSMO can optimize only one Lagrange multiplier at a time when support vectors are unbounded, that is, $\sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} < C$. While decomposition methods such as SMO [6], [15] and SVM-light [16] choose the working set (for example, a maximal violating pair) and optimize it, our method sequentially traverses through the examples without working set selection steps that take time O(|S|). For this case, our method is more efficient than general decomposition methods.

However, when support vectors are bounded, that is, $\sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} = C$, a standard box constraint $0 \leq \sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} \leq C$ has the same effect as a linear equality constraint $\sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} = C$. In this case, the modified FSMO uses the working set selection using second-order information (WSS2) of a practical implementation of SMO [15]. The algorithm chooses $\alpha_{\hat{\mathbf{y}}}$ in $S$ to maximize $g(\hat{\mathbf{y}}) = \Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}})$, and another $\alpha_{\hat{\mathbf{y}}'} > 0$ to minimize $-(g(\hat{\mathbf{y}}) - g(\hat{\mathbf{y}}'))^2 / ||\delta\Psi(\hat{\mathbf{y}}) - \delta\Psi(\hat{\mathbf{y}}')||^2$. Also, $\alpha_{\hat{\mathbf{y}}}$ and $\alpha_{\hat{\mathbf{y}}'}$ are optimized as follows:

$$s_2 = \frac{g(\hat{\mathbf{y}}) - g(\hat{\mathbf{y}}')}{\left\|\delta\Psi(\hat{\mathbf{y}}) - \delta\Psi(\hat{\mathbf{y}}')\right\|^2}, \quad s_2^{\text{clipped}} = \min(\alpha_{\hat{\mathbf{y}}'}, s_2), \tag{11}$$

$$\alpha_{\hat{\mathbf{y}}}^{\text{new}} = \alpha_{\hat{\mathbf{y}}} + s_2^{\text{clipped}}, \quad \alpha_{\hat{\mathbf{y}}'}^{\text{new}} = \alpha_{\hat{\mathbf{y}}'} - s_2^{\text{clipped}}. \tag{12}$$

A pseudocode of the modified FSMO is depicted in algorithm 2. The algorithm is called the 1-slack cutting plane algorithm (line 4 in algorithm 1) and is used to solve the dual problem over the working set $S$. Iterating through the constraint $\hat{\mathbf{y}} = (\hat{y}_1,\cdots,\hat{y}_n)$ in the working set $S$, the algorithm updates

**Algorithm 2**. Modified FSMO algorithm for a 1-slack structural SVM. The algorithm is called the 1-slack cutting plane algorithm (line 4 in algorithm 1) and is used to solve the dual problem over the working set *S*.

1: Input: $(x_1,y_1),\cdots,(x_n,y_n)$, $S$, $\alpha_S$, $C$, $\varepsilon$
2: repeat
3: if $\sum_{\hat{\mathbf{y}}\in Y^n}\alpha_{\hat{\mathbf{y}}} < C$ do /* unbounded SVs: FSMO */
4: for $\hat{\mathbf{y}} = (\hat{y}_1,\cdots,\hat{y}_n) \in S$ do
5: if $\{\Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}}) > \varepsilon,\ \alpha_{\hat{\mathbf{y}}} < C\}$ or $\{\Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}}) < -\varepsilon,\ \alpha_{\hat{\mathbf{y}}} > 0\}$ do
6: calculate $s$ and $s^{\text{clipped}}$
7: $\alpha_{\hat{\mathbf{y}}}^{\text{new}} \leftarrow \alpha_{\hat{\mathbf{y}}} + s^{\text{clipped}}$
8: end if
9: end for
10: else /* bounded SVs: SMO */
11: $\hat{\mathbf{y}} = \arg\max_{\hat{\mathbf{y}}'\in S} g(\hat{\mathbf{y}})$ where $g(\hat{\mathbf{y}}) = \Delta(\hat{\mathbf{y}}) - \mathbf{w}^T\delta\Psi(\hat{\mathbf{y}})$
12: $\hat{\mathbf{y}}' = \arg\min_{\alpha_{\hat{\mathbf{y}}'}>0,\hat{\mathbf{y}}'\in S} \frac{-(g(\hat{\mathbf{y}}) - g(\hat{\mathbf{y}}'))^2}{\|\delta\Psi(\hat{\mathbf{y}}) - \delta\Psi(\hat{\mathbf{y}}')\|^2}$
13: if $g(\hat{\mathbf{y}}) - g(\hat{\mathbf{y}}') > \varepsilon$ do
14: calculate $s_2$ and $s_2^{\text{clipped}}$
15: $\alpha_{\hat{\mathbf{y}}}^{\text{new}} = \alpha_{\hat{\mathbf{y}}} + s_2^{\text{clipped}}$, $\alpha_{\hat{\mathbf{y}}'}^{\text{new}} = \alpha_{\hat{\mathbf{y}}'} - s_2^{\text{clipped}}$
16: end if
17: end if
18: until no $\alpha_{\hat{\mathbf{y}}}$ has changed during iteration

individual Lagrange multipliers (that is, $\alpha_{\hat{\mathbf{y}}}$) and **w** by using (9) and (10) when support vectors are unbounded (lines 4 to 9). When support vectors are bounded, the algorithm chooses two Lagrange multipliers by using the working set selection algorithm of SMO and updates two Lagrange multipliers by using (11) and (12). The algorithm stops if no $\alpha_{\hat{\mathbf{y}}}$ has changed during iteration.

### 1. Convergence of the Modified FSMO Algorithm

Let us now discuss the convergence of algorithm 2. The core idea is to show that the improvement of the dual objective in (6), $D(\alpha^{k+1}) - D(\alpha^k)$, can be lower bounded by a positive constant [17]. We will prove that there exists $\sigma > 0$ such that

$$\forall k : D(\alpha^{k+1}) - D(\alpha^k) \ge \frac{\sigma}{2}\left\|\alpha^{k+1} - \alpha^k\right\|^2. \tag{13}$$

First, we assume that support vectors are unbounded. Let the parametric change in $\alpha$ be given by $\alpha(t)$:

$$\alpha_{\hat{\mathbf{y}}}(t) \equiv \alpha_{\hat{\mathbf{y}}}^k + t, \qquad \alpha_{\bar{\mathbf{y}}}(t) \equiv \alpha_{\bar{\mathbf{y}}}^k, \forall \bar{\mathbf{y}} \ne \hat{\mathbf{y}}, \tag{14}$$

where $\hat{\mathbf{y}}, \bar{\mathbf{y}} \in S$ are violating elements.

The subproblem is to maximize $D(t)$. Let $\bar{t}$ denote the solution of this problem, and $\alpha^{k+1} = \alpha(\bar{t})$. Clearly,

$$t = \left\|\alpha^{k+1} - \alpha^k\right\|. \tag{15}$$

As $D(t)$ is a quadratic function on $t$,

$$D(t) = D(0) + D'(0)t + D''(0)\frac{t^2}{2}. \tag{16}$$

Since

$$D'(t) = \Delta(\hat{\mathbf{y}}) - \sum_{\bar{\mathbf{y}}\in Y^n}\alpha_{\bar{\mathbf{y}}}\delta\Psi(\bar{\mathbf{y}})^T\delta\Psi(\hat{\mathbf{y}}),$$
$$D''(t) = -\delta\Psi(\hat{\mathbf{y}})^T\delta\Psi(\hat{\mathbf{y}}),$$

we have

$$D''(0) = -\left\|\delta\Psi(\hat{\mathbf{y}})\right\|^2. \tag{17}$$

Let $t^*$ denote the unconstrained minimum of $D$, that is, $t^* = -D'(0)/D''(0)$. Clearly, $\bar{t} = \gamma t^*$, where $0 < \gamma \le 1$. Then, by (15) through (17),

$$D(t) - D(0) = \gamma\frac{-D'(0)^2}{D''(0)} + \frac{\gamma^2}{2}\frac{D'(0)^2}{D''(0)}$$
$$\ge -\frac{\gamma^2}{2}\frac{D'(0)^2}{D''(0)} = -\frac{D''(0)}{2}\bar{t}^2 = \frac{\left\|\delta\Psi(\hat{\mathbf{y}})\right\|^2}{2}\left\|\alpha^{k+1} - \alpha^k\right\|^2. \tag{18}$$

In a case in which support vectors are bounded, we can obtain

$$D(t) - D(0) \ge \frac{\left\|\delta\Psi(\hat{\mathbf{y}}) - \delta\Psi(\bar{\mathbf{y}})\right\|^2}{4}\left\|\alpha^{k+1} - \alpha^k\right\|^2. \tag{19}$$

In both cases, $\{\alpha_k\}$ converges to some $\bar{\alpha}$ [18].

Since $\left\|\delta\Psi(\hat{\mathbf{y}})\right\|^2 \ge \left\|\delta\Psi(\hat{\mathbf{y}}) - \delta\Psi(\bar{\mathbf{y}})\right\|^2/2$ and algorithm 2 does not require working set selection algorithms that take time O(|S|) when support vectors are unbounded, algorithm 2 converges faster than general SMO-type decomposition methods such as SVM-light.

## IV. Related Works

We now compare our method with other methods. Platt briefly introduced a fixed-threshold SMO (FSMO) for a fixed-threshold SVM where the bias $b$ is fixed at zero but did not apply it to structural SVMs [6].

The 1-slack cutting plane method of Joachims [13] and the bundle method of Teo [12] are given for the general setting of SVMs with structured outputs, but they use a standard SVM solver (SVM-light) to solve the dual form of structural SVMs, despite the fact that a structural SVM has no bias, that is, the bias $b$ is fixed at zero. To speed up the convergence of the cutting plane method, Franc and Sonnenburg proposed a new

method called the optimized cutting plane algorithm (OCAS) [19]. Unlike standard cutting plane methods, OCAS aims at simultaneously optimizing the master and reducing the problem's objective functions in (1) and (2) using a line-search. It will be interesting to apply OCAS to improve the speed of the 1-slack cutting plane algorithm (algorithm 1) in our work.

Keerthi presented sequential descent methods (SDM) for the Crammer-Singer and Weston-Watkins multiclass linear SVM formulations [20]. Like our method, their method updates the dual variables associated with one example at a time. However, they did not apply SDM to a structural SVM. Their method is not easily extendable to a structural SVM because the QP subproblem associated with each example does not have a simple form such as those in multiclass SVM formulations, and some care is needed to solve this subproblem efficiently [20].

The exponentiated gradient (EG) method [8] also applies to structured output problems. The EG method also updates the dual variables associated with one example at a time, but the selection of examples is done in an online mode where $i$ is picked randomly each time.

Stochastic subgradient descent methods were proposed for structured output problems [9]. The Pegasos algorithm showed promising performance for binary classification SVMs [10]; however, this algorithm lacks a good stopping criterion.

For comparison, we extend the Pegasos algorithm for structured prediction (Pegasos-struct). We replace the objective of Pegasos with an approximate objective function of $n$-slack structural SVMs,

$$f(\mathbf{w};A_t)=\frac{\lambda}{2}\mathbf{w}^T\mathbf{w}+\frac{1}{k}\sum_{(x,y)\in A_t} l(\mathbf{w};(x,y)), \quad (20)$$

$$\text{where } l(\mathbf{w};(x,y))=\max_{\hat{y}}\left\{\Delta(y,\hat{y})-\mathbf{w}^T\delta\Psi(x,\hat{y})\right\}.$$

The parameter $k$ is the number of examples used for calculating subgradients, and $A_t$ is the subset of a training set $S$. The subgradient of $f(\mathbf{w};A_t)$ is

$$\nabla f(\mathbf{w};A_t)=\lambda\mathbf{w}-\frac{1}{|A_t|}\sum_{(x,y)\in A_t^+}\delta\Psi(x,\hat{y}), \quad (21)$$

$$\text{where } \hat{y}=\arg\max_{\hat{y}}\left\{\Delta(y,\hat{y})-\mathbf{w}^T\delta\Psi(x,\hat{y})\right\}.$$

In the previous equation, $A_t^+$ is the set of examples for which $\mathbf{w}$ suffers a non-zero loss. A pseudocode of the modified Pegasos is given in algorithm 3.

## V. Experiments

We implemented 1-slack structural SVMs using a modified FSMO in C++. For comparison, we ran an $n$-slack FSMO [11]

**Algorithm 3.** Modified Pegasos algorithm (Pegasos-struct) for $n$-slack structural SVMs.

1: Input: $S, \lambda, T, k$

2: Initialize: Choose $\mathbf{w}_1$ s.t. $\|\mathbf{w}_1\|\le 1/\sqrt{\lambda}$

3: for $t$=1,2,⋯,$T$ do

4: Choose $A_t\subseteq S$, where $|A_t|=k$

5: Set $A_t^+=\left\{(x,y)\in A_t:\Delta(y,\hat{y})>\mathbf{w}^T\delta\Psi(x,\hat{y})\right\}$

6: Set $\eta_t=1/\lambda t$

7: Set $\mathbf{w}_{t+1/2}=(1-\eta_t\lambda)\mathbf{w}_t+\frac{\eta_t}{k}\sum_{(x,y)\in A_t^+}\delta\Psi(x,\hat{y})$

8: Set $\mathbf{w}_{t+1}=\min\left\{1,\frac{1/\sqrt{\lambda}}{\|\mathbf{w}_{t+1/2}\|}\right\}\mathbf{w}_{t+1/2}$

9: end for

10: Output: $\mathbf{w}_{T+1}$

and an $n$-slack and 1-slack SVM-struct that uses SVM-light for solving QP problems [13]. We also implemented Pegasos-struct for the $n$-slack structural SVM by modifying the Pegasos algorithm [10] and compared it to other structural SVM algorithms.

For all structural SVMs, a linear kernel was used. Regularization constant $C$ from {10, 100, 1000, 10000} was chosen by optimization of the test set for all experiments. As the stopping condition precision, we set $\varepsilon$ = 0.01 for multiclass classification tasks and $\varepsilon$ = 0.1 for sequence labeling tasks. For Pegasos-struct, the parameters $\lambda$ and $k$ are used as $\lambda=1/C$[1)] and $k$=1, respectively. We used the stopping condition $|1-f(\mathbf{w}_{t+1})/f(\mathbf{w}_t)|\le 0.01$ for Pegasos-struct. The experiments were performed on a 2.5 GHz Intel Core2 Quad CPU machine using Windows XP. The following datasets were used for various experiments: MNIST [21], NEWS20 [22], CoNLL-2000

Table 1. Data sets used in the experiments.

| Data set | Training size | Test size | Class | Feature | Parameters |
|---|---|---|---|---|---|
| MNIST | 60,000 | 10,000 | 10 | 780 | C=1000, $\varepsilon$=0.01 |
| NEWS20 | 15,935 | 3,993 | 20 | 62,061 | C=100, $\varepsilon$=0.01 |
| CoNLL-2000 chunking | 211,727 | 47,377 | 22 | 387,875 | C=1000, $\varepsilon$=0.1 |
| Korean spacing | 987,869 | 329,219 | 2 | 228,260 | C=1000, $\varepsilon$=0.1 |

1) For the regularization parameter $C$ and λ, we use the following relation: $\lambda=1/C$.

Table 2. Training time and performance of data sets.

| Data set | Algorithm | Training time (s) | F1 (%) |
|---|---|---|---|
| MNIST | 1-slack FSMO | 152 | 92.47 |
| | *n*-slack FSMO | 286 | 92.50 |
| | Pegasos-struct | 34 | 91.88 |
| | 1-slack SVM-struct | 321 | 92.39 |
| | *n*-slack SVM-struct | 5,797 | 92.37 |
| NEWS20 | 1-slack FSMO | 41 | 84.62 |
| | *n*-slack FSMO | 81 | 84.52 |
| | Pegasos-struct | 11 | 84.40 |
| | 1-slack SVM-struct | 69 | 84.55 |
| | *n*-slack SVM-struct | 3,084 | 84.57 |
| CoNLL-2000 chunking | 1-slack FSMO | 679 | 93.75 |
| | *n*-slack FSMO | 1,449 | 93.80 |
| | Pegasos-struct | 67 | 93.69 |
| | 1-slack SVM-struct | 1,189 | 93.77 |
| | *n*-slack SVM-struct | 30,184 | 93.77 |
| Korean spacing | 1-slack FSMO | 771 | 97.04 |
| | *n*-slack FSMO | 1,662 | 97.01 |
| | Pegasos-struct | 50 | 96.63 |
| | 1-slack SVM-struct | 2,005 | 97.05 |
| | *n*-slack SVM-struct | 118,492 | 97.04 |

[23], and Korean spacing data set [24]. For NEWS20, MNIST, and CoNLL-2000, we used the standard train-test split. For training of Korean spacing, we used about 1.3% of the 21st Century Sejong Project's raw corpus.[2] For evaluation of Korean spacing, we used the ETRI POS tagged corpus [24]. Table 1 summarizes the characteristics of the data sets and parameters for structural SVMs.

Table 2 shows the training time and test set performance on all data sets. The modified FSMO for the 1-slack structural SVM (1-slack FSMO) outperforms *n*-slack SVM-struct, 1-slack SVM-struct, and *n*-slack FSMO on all data sets in terms of training time while obtaining a comparable performance of the test set. Pegasos-struct is fast, but it fails to achieve a precise solution on the Korean spacing data set.

Figures 2 through 5 show log-log plots of how training times increase with the size of the training set on the MNIST data set (multiclass classification task), NEWS20 data set (multiclass classification task), CoNLL-2000 chunking data set (sequence labeling task), and Korean spacing data set (sequence labeling task), respectively. The lines in a log-log plot correspond to

2) http://www.sejong.or.kr/eindex.php

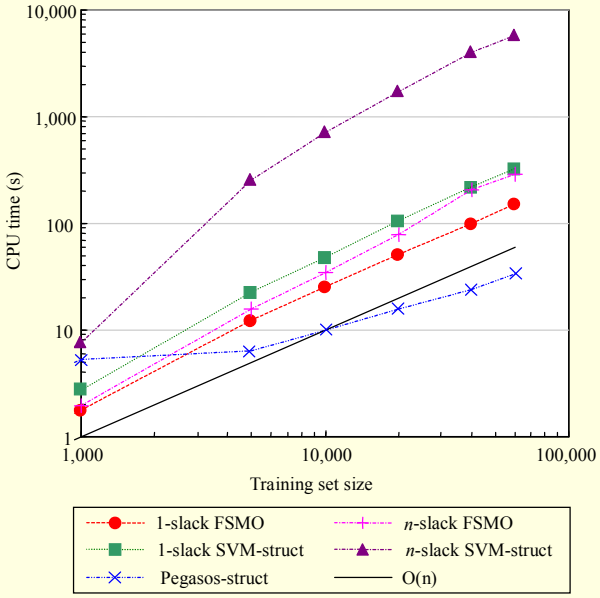


Fig. 2. Training times of the MNIST data set.

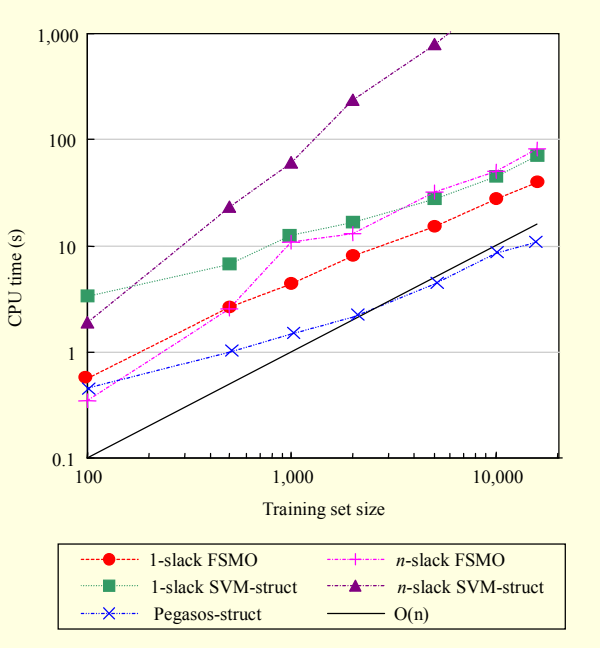


Fig. 3. Training times of the NEWS20 data set.

polynomial growth $O(n^d)$, where $d$ corresponds to the slope of the line. The proposed method (1-slack FSMO) is faster than *n*-slack SVM-struct, 1-slack SVM-struct, and *n*-slack FSMO on all data sets. Pegasos-struct is fast on large training set sizes, but it often fails to achieve a precise solution.

Figures 6 through 9 show the primal objective values and dual objective values on the MNIST data set, the NEWS20 data set, CoNLL-2000 data set, and Korean spacing data set, respectively. The proposed method (1-slack FSMO)

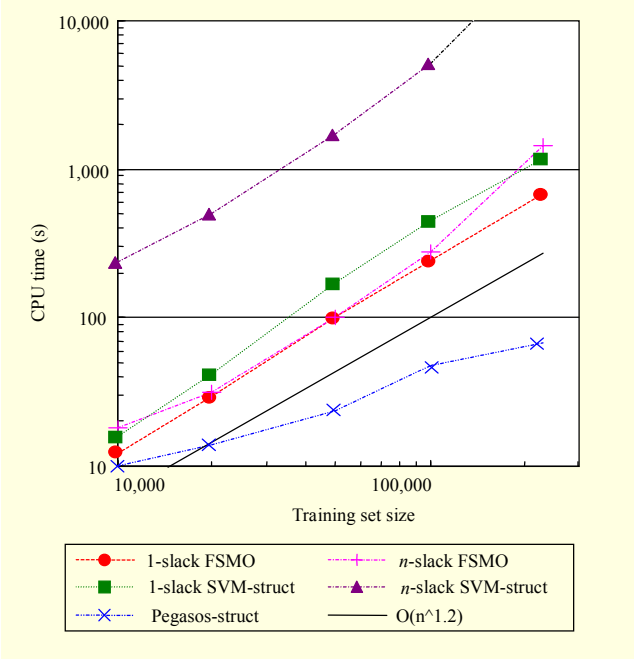


Fig. 4. Training times of the CoNLL-2000 data set.

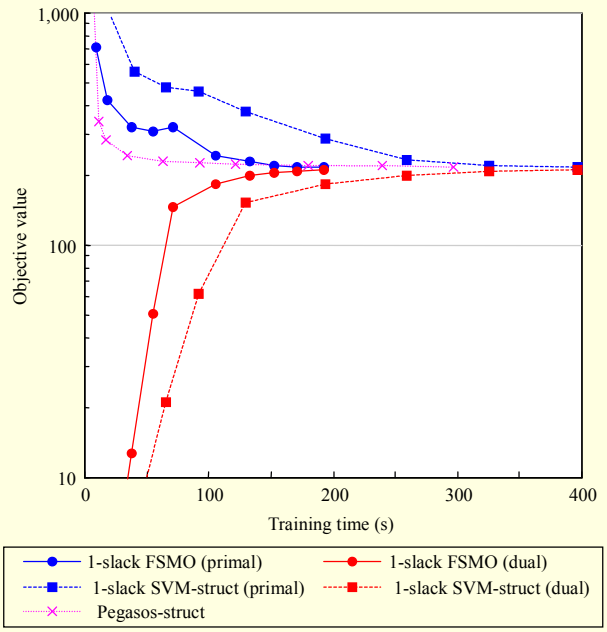


Fig. 6. Objective values of the MNIST data set.

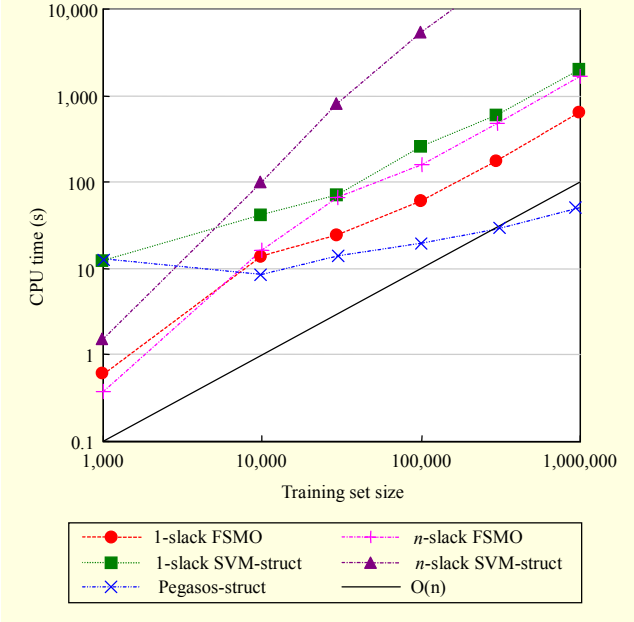


Fig. 5. Training times of the Korean spacing data set.

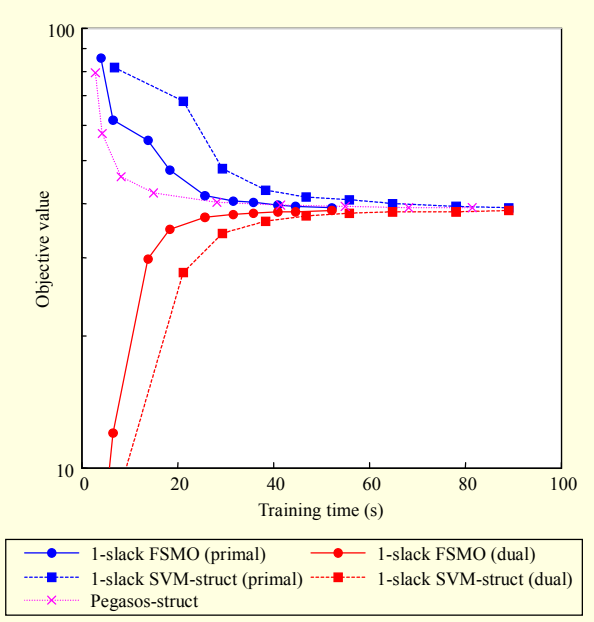


Fig. 7. Objective values of the NEWS20 data set.

consistently outperforms 1-slack SVM-struct on all data sets in terms of training time while obtaining a comparable objective value. Pegasos-struct is fast in the beginning, but it often fails to achieve a precise solution.

## VI. Conclusion

This paper presents a modified FSMO for 1-slack structural SVMs. The modified FSMO is conceptually simple, easy to implement, and faster than the standard SVM training algorithms for 1-slack structural SVMs. For various experiments, the proposed method is faster than *n*-slack SVM-struct, 1-slack SVM-struct, and *n*-slack FSMO without decreasing the performance. Pegasos-struct is fast on large training set sizes, but it fails to achieve a precise solution on some data sets.

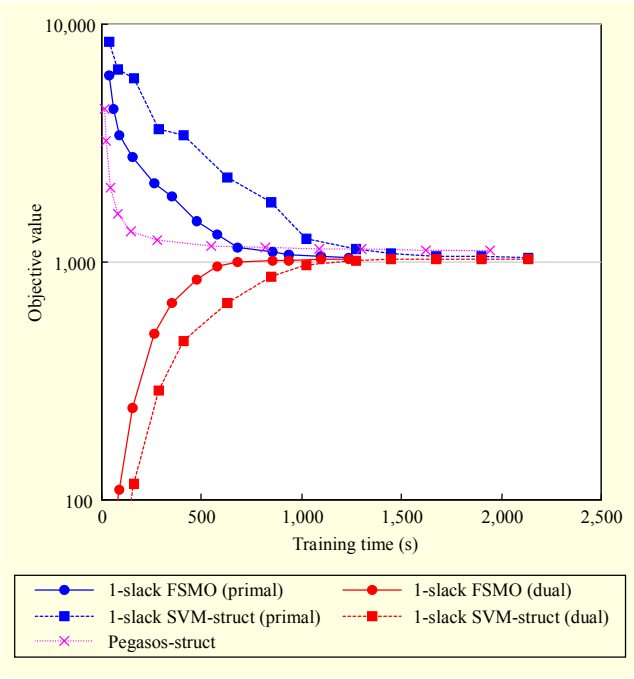


Fig. 8. Objective values of the CoNLL-2000 data set.

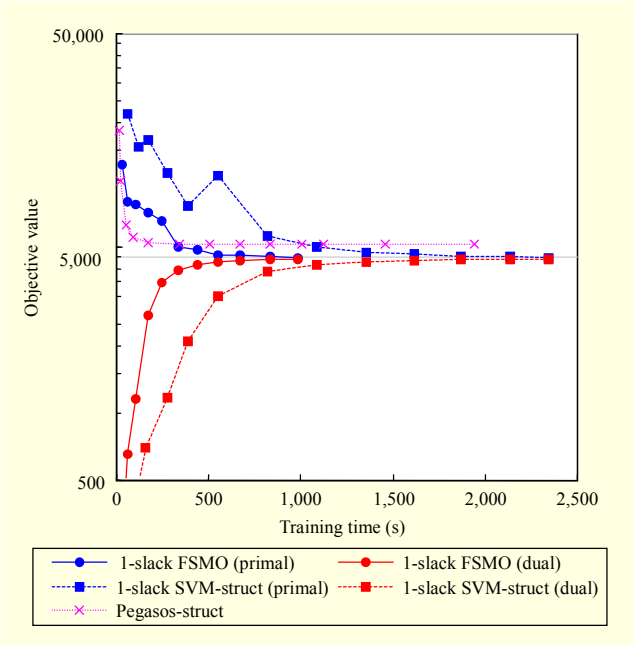


Fig. 9. Objective values of the Korean spacing data set.

## References


[1] B. Taskar, C. Guestrin, and D. Koller, "Max Margin Markov Networks," *NIPS*, vol. 16, 2004.

[2] I. Tsochantaridis et al., "Support Vector Machine Learning for Interdependent and Structured Output Spaces," *Proc. ICML*, 2004, pp. 104.

[3] B. Taskar et al., "Max-Margin Parsing," *Proc. EMNLP*, 2004.

[4] C.N. Yu et al., "Support Vector Training of Protein Alignment Models," *Proc. RECOMB*, 2007, pp. 253-267.

[5] Y. Yue et al., "A Support Vector Method for Optimization Average Precision," *Proc. SIGIR*, 2007, pp. 271-278.

[6] J. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," Microsoft Research Technical Report MSR-TR-98-14, 1998.

[7] H. Kim and W. Kim, "Eye Detection in Facial Images Using Zernike Moments with SVM," *ETRI J.*, vol. 30, no. 2, 2008, pp. 335-337.

[8] M. Collins et al., "Exponentiated Gradient Algorithms for Conditional Random Fields and Max-Margin Markov Networks," *JMLR*, vol. 9, 2008, pp. 1775-1822.

[9] N.D. Ratliff, J.A. Bagnell, and M.A. Zinkevich, "(Online) Subgradient Methods for Structured Prediction," *Proc. AISTATS*, 2007.

[10] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal Estimated Subgradient Solver for SVM," *Proc. ICML*, 2007, pp. 807-814.

[11] C. Lee and M. Jang, "Fast Training of Structured SVM Using Fixed-Threshold Sequential Minimal Optimization," *ETRI J.*, vol. 31, no. 2, 2009, pp. 121-128.

[12] C.H. Teo et al., "A Scalable Modular Convex Solver for Regularized Risk Minimization," *Proc. KDD*, 2007, pp. 727-736.

[13] T. Joachims, T. Finley, and C.N. Yu, "Cutting-Plane Training of Structural SVMs," *MLJ*, vol. 77, no. 1, 2009, pp. 27-59.

[14] T. Joachims, "Training Linear SVMs in Linear Time," *Proc. KDD*, 2006, pp. 217-226.

[15] R. Fan, P. Chen, and C. Lin, "Working Set Selection Using Second Order Information for Training Support Vector Machines," *JMLR*, vol. 6, 2005, pp. 1889-1918.

[16] T. Joachims, "A Statistical Learning Model of Text Classification with Support Vector Machines," *Proc. SIGIR*, 2001, pp. 128-136.

[17] C. Lin, "Asymptotic Convergence of an SMO Algorithm Without Any Assumptions," *IEEE Trans. Neural Networks*, vol. 13, no. 1, 2002, pp. 248-250.

[18] S. Keerthi and E. Gilbert, "Convergence of a Generalized SMO Algorithm for SVM Classifier Design," *Machine Learning*, vol. 46, no. 1-3, 2002, pp. 351-360.

[19] V. Franc and S. Sonnenburg, "Optimized Cutting Plane Algorithm for Support Vector Machines," *Proc. ICML*, vol. 307, 2008, pp. 320-327.

[20] S. Keerthi et al. "A Sequential Dual Method for Large Scale Multi-Class Linear SVMs," *Proc. KDD*, 2008, pp. 408-416.

[21] Y. LeCun et al., "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, Nov. 1988, pp. 2278-2324.

[22] K. Lang. "Newsweeder: Learning to Filter Netnews," *Proc. ICML*, 1995.

[23] E.F.T.K. Sang and S. Buchholz. "Introduction to the CoNLL-2000 Shared Task: Chunking," *Proc. CoNLL-2000 and LLL-2000*, 2000, pp. 127-132.

[24] D. Lee, H. Rim, and D. Yook, "Automatic Word Spacing Using Probabilistic Models Based on Character n-Grams," *IEEE Intelligent Systems*, vol. 22, no. 1, 2007, pp. 28-35.

**Changki Lee** received the BS degree in computer science from KAIST, Korea, in 1999. He received the MS and PhD degrees in computer engineering from POSTECH, Korea, in 2001 and 2004, respectively. Since 2004, he has been with Electronics and Telecommunications Research Institute (ETRI), Daejeon, Korea, as a senior member of research staff. He has served as a reviewer for some international journals, such as *Information System*, *Information Processing and Management*, and *ETRI Journal*. His research interests are natural language processing, information retrieval, data mining, and machine learning.

**Myung-Gil Jang** received the BS and MS degrees in computer science and statistics from Pusan National University, Korea, in 1988 and 1990, respectively. He received the PhD degree in information science from Chungnam National University in 2002. He was with System Engineering Research Institute (SERI), Korea, from 1990 to 1997 as a researcher. Since 1998, he has been with Electronics and Telecommunications Research Institute (ETRI), Korea, as a senior/principle member of research staff. His research interests are natural language processing, information retrieval, question answering, knowledge and dialogue processing, media retrieval/management, and semantic webs.