Structured Output Learning with Candidate Labels for Local Parts: Supplementary Materials

Chengtao Li[1], Jianwen Zhang[2], and Zheng Chen[2]

[1] Institute for Interdisciplinary Information Sciences, Tsinghua University, Beijing, China, 100080
lichengtao2010@gmail.com,
[2] Microsoft Research Asia, Beijing, China, 100080
jiazhan@microsoft.com, zhengc@microsoft.com

1 Proof of Theorem 1

Theorem 1. *Given a structured instance* $\mathbf{x}$ *and arbitrary candidate labeling set* Y*, no algorithm exists that can always find the most violated label (in* Y *or not in* Y*) in poly(*$|\mathbf{x}|$*) time unless* $P = NP$*, where* $|\mathbf{x}|$ *is the length of* $\mathbf{x}$.

Sketch of the Proof. We prove this theorem by first proving the following lemmas:

Lemma 1. *We prove that no algorithm exists that can always find the most violated label setting that is in* Y *where* Y *could be an arbitrary candidate label set.*

Lemma 2. *We prove that no algorithm exists that can always find the most violated label setting that is in* $\mathcal{Y}/Y$ *where* Y *could be arbitrary candidate label set.*

By combining these two lemmas we finish the proof of the theorem.

Proof. Lemma 1: Assume that for arbitrary candidate label set Y, an algorithm exists that can find the most violated label setting that is in Y in $poly(|\mathbf{x}|)$ time.

The value of $|Y|$ can be exponential in $|\mathbf{x}|$, without proper encoding of the candidate label set Y, it would take $\exp(|\mathbf{x}|)$ time to read Y. So if the algorithm runs in $poly(|\mathbf{x}|)$ time, there must exist some kind of encoding of the candidate label set and the given Y is already encoded. Thus we show that even with encoding, the algorithm still cannot run in $poly(|\mathbf{x}|)$.

Assume that given Y is encoded in the following rules:
Rule 1: For all label settings in Y*, at least one of the following cases happens: the label of* x_i *is* y_i*; or the label of* x_j *is* y_j*, or* $\ldots$
Rule 2: For all label settings in Y*, at least one of the following cases happens: the label of* x_k *is* y_k*; or* $\ldots$
$\ldots$

Now we prove that finding the most violated label setting in Y is *NP-hard*. More precisely, we prove that the decision version of this problem: determining whether a label setting exists that is in Y is *NP-complete*.

It is obvious that this decision problem is in *NP*, since given a label setting we could determine whether it is in Y by checking all the rules in $poly(|\mathbf{x}|)$.

Now we use the solver of this problem as a black box, and prove that *3-SAT* problem could be reduced to this problem in polynomial time. Given any instance of *3-CNF* with variables $x_1, \ldots, x_N$, say,

$$\phi = (x_{i_1} \vee \overline{x_{j_1}} \vee x_{k_1}) \wedge (\overline{x_{i_2}} \vee x_{j_2} \vee x_{k_2}) \wedge \ldots \quad (1)$$

To determine whether it is satisfiable, we construct 2 kinds of labels $true$ and $false$, and a set of variables $z_1, \ldots, z_N$, and want to see whether there exists a label setting of $(z_1, \ldots, z_N)$ that is in the candidate label set Y. To construct Y, we encode ϕ into the rules in it. For example, a clause $(x_{i_n} \vee \overline{x_{j_n}} \vee x_{k_n})$ is encoded into the following rule:
Rule n: For all label settings in Y, at least one of the following cases happen: the label of x_{i_n} is true; the label of x_{j_n} is $false$; the label of x_{k_n} is true

This encoding only needs polynomial time in N if the encoding of ϕ itself is $poly(|N|)$. And it is obvious that the black box will return "yes" (which means, there exists a label setting exists that meets all the rules in Y) if and only if ϕ is satisfiable.

Hence, this problem is *NP-Complete*.

Lemma 2: This time we assume that Y is encoded in the following rule:
Rule: For all label settings in Y, at least one of the following cases happens:
Case 1: The label of x_i is y_i, and the label of x_j is y_j, and ...
Case 2: The label of x_k is y_k; and ...
...
and the decision version of this problem becomes: determining whether a label setting exists in $\mathcal{Y}/Y$.

We know that determining whether a *3-DNF* problem is unsatisfiable is also *NP-Complete*, and with a similar proof we could also show that it could be reduced to this problem in polynomial time, indicating that this problem is *NP-Complete*, which finishes the proof.

2 Proof of Theorem 2

Theorem 2. *If the candidate labels are given marginally by local parts, namely, each Y_i in $\{\mathbf{x}_i, Y_i\}_{i=1}^N$ has the form $Y_i = \{Y_{i1} \otimes Y_{i2} \otimes \ldots \otimes Y_{iM_i}\} \subseteq \mathcal{Y}$, where Y_{ij} is the set of candidate labels that $\mathbf{x}_{ij}$ could possibly take, among which only one is fully correct; $\mathbf{x}_{ij}$ is the j-th local part in $\mathbf{x}_i$ whose size is upper bounded by some constant; M_i is the number of local parts in $\mathbf{x}_i$, then in the sequence structured learning an efficient algorithm exists (modified Viterbi algorithm) that could find the most violated candidate/non-candidate labels.*

Proof. We show the algorithms obtained by slightly modifying the *Viterbi* algorithm, that could find the most violated candidate label setting and non-candidate label setting in Algorithm 1 and Algorithm 2 respectively. Note that

Algorithm 1 Viterbi for finding the most violated candidate label setting

Input: Transition Weight Matrix W_T, Emission Weight Vector W_E, Structured Instance $\mathbf{x}$, corresponding candidate label set Y
Output: The most violated candidate label setting Z

```
t ← |x|
for each i ∈ Y_1 do
    T_1[i, 1] ← W_E[i]
    T_2[i, 1] ← i
end for
for i ← 2, 3, ..., t do
    for each j ∈ Y_i do
        T_1[j, i] ← max_{k∈Y_{i−1}} {T_1[k, i − 1] + W_T[k, j] + W_E[j]}
        T_2[j, i] ← arg max_{k∈Y_{i−1}} {T_1[k, i − 1] + W_T[k, j] + W_E[j]}
    end for
end for
Z[t] ← arg max_{k∈Y_t} T_1[k, t]
for i = t to 2 do
    Z[i − 1] ← T_2[Z[i], i]
end for
```

we assume the candidate labels are given token-wisely, but it's easy to be generalized to the case where candidate labels are given marginally.

It is obvious that the time complexity of these two modified Viterbi algorithms are of the same scale, i.e., $O(n*T^2)$ where n is the length of the sequence, T is the size of the label space.

This time complexity is the same as the original Viterbi algorithm, and is polynomial in the length of sequence n and the number of labels T. Thus these two algorithms can efficiently find the most violated candidate/non-candidate label setting in a sequence.

3 Proof of Theorem 3

Theorem 3. *$\forall \mathbf{w}, \mathcal{J}_0(\mathbf{w}) \geq \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) \geq \mathcal{J}_m(\mathbf{w})$ and*

$$\mathcal{J}_0^* \geq \mathcal{J}_c^* \geq \mathcal{J}_m^*.$$

Proof. The true problem of supervised learning if we know the true labels $\mathbf{y}_i^*$'s:

$$\begin{aligned}\min_{\mathbf{w}} \mathcal{J}_0(\mathbf{w}) = &\sum_{i=1}^N C_1 \left| \max_{\mathbf{y}_i' \in Y_i} [\Delta(\mathbf{y}_i^*, \mathbf{y}_i') + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i', \mathbf{y}_i^*) \rangle] \right|_+ \\ &+ \sum_{i=1}^N C_2 \left| \max_{\mathbf{y}_i'' \in \mathcal{Y}/Y_i} [\Delta(\mathbf{y}_i^*, \mathbf{y}_i'') + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i'', \mathbf{y}_i^*) \rangle] \right|_+ + \frac{1}{2}\|\mathbf{w}\|^2. \quad (2)\end{aligned}$$

Algorithm 2 Viterbi for finding the most violated non-candidate label setting

Input: Transition Weight Matrix W_T, Emission Weight Vector W_E, Structured Instance $\mathbf{x}$, corresponding candidate label set Y, number of classes N
Output: The most violated candidate label setting Z
$t \leftarrow |\mathbf{x}|$
for each $i \in Y_1$ **do**
  $T_1[i, 1] \leftarrow W_E[i]$
  $T_2[i, 1] \leftarrow i$
end for
for each $i \in [N]/Y_1$ **do**
  $T_1'[i, 1] \leftarrow W_E[i]$
  $T_2'[i, 1] \leftarrow i$
end for
for $i \leftarrow 2, 3, \ldots, t$ **do**
  for each $j \in Y_i$ **do**
    $T_1[j, i] \leftarrow \max_{k \in Y_{i-1}} \{T_1[k, i-1] + W_T[k, j] + W_E[j]\}$
    $T_2[j, i] \leftarrow \arg\max_{k \in Y_{i-1}} \{T_1[k, i-1] + W_T[k, j] + W_E[j]\}$
  end for
  for each $j \in [N]/Y_i$ **do**
    $T_1'[j, i] \leftarrow \max_k \{T_1[k, i-1] + W_T[k, j] + W_E[j]\}$
    $T_2[j, i] \leftarrow \arg\max_k \{T_1[k, i-1] + W_T[k, j] + W_E[j]\}$
  end for
end for
$Z[t] \leftarrow \arg\max_{k \in [N]/Y_t} T_1'[k, t]$
for $i = t$ **to** 2 **do**
  $Z[i-1] \leftarrow T_2[Z[i], i]$
end for

The problem of CLLP:

$$\min_{\mathbf{w}, \{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) = \sum_{i=1}^N C_1 \left| \max_{\mathbf{y}_i' \in Y_i} [\Delta(\mathbf{y}_i, \mathbf{y}_i') + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i', \mathbf{y}_i) \rangle] \right|_+ + \sum_{i=1}^N C_2 \left| \max_{\mathbf{y}_i'' \in \mathcal{Y}/Y_i} [\Delta(\mathbf{y}_i, \mathbf{y}_i'') + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i'', \mathbf{y}_i) \rangle] \right|_+ + \frac{1}{2}\|\mathbf{w}\|^2, \quad (3)$$

The problem of MMS:

$$\min_{\mathbf{w}} \mathcal{J}_m = C_2 \sum_{i=1}^N \left| \max_{\mathbf{y}_i'' \notin Y_i} [\Delta_{\min}(\mathbf{y}_i'', \mathcal{Y}/Y_i) + \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i'') \rangle] - \max_{\mathbf{y}_i \in Y_i} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle \right|_+ + \frac{1}{2}\|\mathbf{w}\|^2 \quad (4)$$

where $\Delta_{\min}(\mathbf{y}', Y) = \min_{\mathbf{y} \in Y} \Delta(\mathbf{y}', \mathbf{y})$

Lemma 3. *$\forall \mathbf{w}$, $\mathcal{J}_0(\mathbf{w}) \geq \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N)$. Namely, the objective Equation 2 upper bounds the objective Equation 3.*

Proof. $\mathbf{y}_i^* \in Y_i \Rightarrow \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) \leq \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i^*\}_{i=1}^N) = \mathcal{J}_0(\mathbf{w})$.

Corollary 1. *Let* $\mathcal{J}_0^* = \min_{\mathbf{w}} \mathcal{J}_0(\mathbf{w})$, *and* $\mathcal{J}_c^* = \min_{\mathbf{w}, \{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N)$, *then* $\mathcal{J}_0^* \geq \mathcal{J}_c^*$. *Namely, the optimal value of the objective Equation 2 upper bounds that of the objective Equation 3.*

Proof. Let $\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathcal{J}_0(\mathbf{w})$, *then*

$$\mathcal{J}_0^* = \mathcal{J}_0(\mathbf{w}^*) \geq \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}^*, \{\mathbf{y}_i\}_{i=1}^N) \geq \min_{\mathbf{w}, \{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) = \mathcal{J}_c^*.$$

Lemma 4. $\forall \mathbf{w}, \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) \geq \mathcal{J}_m(\mathbf{w})$. *Namely, the objective Equation 3 upper bounds the objective Equation 4.*

Corollary 2. *Let* $\mathcal{J}_c^* = \min_{\mathbf{w}, \{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N)$, *and* $\mathcal{J}_m^* = \min_{\mathbf{w}} \mathcal{J}_m(\mathbf{w})$, *then* $\mathcal{J}_c^* \geq \mathcal{J}_m^*$. *Namely, the optimal value of the objective Equation 3 upper bounds that of the objective Equation 4.*

The proofs are similar to those for Lemma 1 and Corolary 1.

By combining the above lemmas and corollaries, we obtain the theorem: $\forall \mathbf{w}, \mathcal{J}_0(\mathbf{w}) \geq \min_{\{\mathbf{y}_i \in Y_i\}_{i=1}^N} \mathcal{J}_c(\mathbf{w}, \{\mathbf{y}_i\}_{i=1}^N) \geq \mathcal{J}_m(\mathbf{w})$ *and*

$$\mathcal{J}_0^* \geq \mathcal{J}_c^* \geq \mathcal{J}_m^*.$$

4 The 2-slack Cutting Plane Algorithm

4.1 Formulation

The formulation of the 2-slack optimization problem:

$$\begin{aligned}
&\min_{\mathbf{w},\xi,\zeta} \frac{1}{2}\|\mathbf{w}\|^2 + C_1\xi + C_2\zeta \\
&s.t.\ \forall(\mathbf{y}_1, \ldots, \mathbf{y}_n) \in (Y_1, \ldots, Y_N) \\
&\qquad \xi \geq \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i, \overline{\mathbf{y}_i})\rangle) \\
&\qquad \forall(\mathbf{y}'_1, \ldots, \mathbf{y}'_N) \in (\mathcal{Y}/Y_1 \cup \{\overline{\mathbf{y}_1}\}, \ldots, \mathcal{Y}/Y_N \cup \{\overline{\mathbf{y}_N}\}) \\
&\qquad \zeta \geq \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}'_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}'_i, \overline{\mathbf{y}_i})\rangle)
\end{aligned} \tag{5}$$

4.2 Algorithm

The algorithm is described in Algorithm 3.

Algorithm 3 The 2-Slack Cutting Plane Algorithm

Input: $\{\mathbf{x}_i, Y_i, \overline{\mathbf{y}_i}\}_{i=1}^N$, C_1, C_2, ε_1, ε_2
Initialize $\Omega_1 \leftarrow \emptyset$, $\Omega_2 \leftarrow \emptyset$, $f = true$
repeat
  $f = true$
  $(\mathbf{w}, \xi, \zeta) \leftarrow \arg\min_{\mathbf{w},\xi\geq 0,\zeta\geq 0} \frac{1}{2}\|\mathbf{w}\|^2 + C_1\xi + C_2\zeta$
  s.t. $\forall(\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \Omega_1$:
    $\xi \geq \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i, \overline{\mathbf{y}_i})\rangle)$
    $\forall(\mathbf{y}'_1, \ldots, \mathbf{y}'_N) \in \Omega_2$:
    $\zeta \geq \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}'_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}'_i, \overline{\mathbf{y}_i})\rangle)$
  for $i = 1$ **to** N **do**
    $\mathbf{y}_i \leftarrow \arg\max_{\mathbf{y}_i \in Y_i}\{\Delta(\overline{\mathbf{y}_i}, \mathbf{y}_i) + \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i)\rangle\}$ (modified Viterbi to find the most violated candidate labels)
    $\mathbf{y}'_i \leftarrow \arg\max_{\mathbf{y}_i \in \mathcal{Y}/Y_i \cup \{\overline{\mathbf{y}_i}\}}\{\Delta(\overline{\mathbf{y}_i}, \mathbf{y}'_i) + \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}'_i)\rangle\}$ (modified Viterbi to find the most violated non-candidate labels)
  end for
  if $\xi + \epsilon_1 < \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i, \overline{\mathbf{y}_i})\rangle)$ **then**
    $\Omega_1 \leftarrow \Omega_1 \cup \{(\mathbf{y}_1, \ldots, \mathbf{y}_N)\}$
    $f = false$
  end if
  if $\zeta + \epsilon_2 < \frac{1}{N}\sum_{i=1}^N (\Delta(\mathbf{y}'_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}'_i, \overline{\mathbf{y}_i})\rangle)$ **then**
    $\Omega_2 \leftarrow \Omega_2 \cup \{(\mathbf{y}'_1, \ldots, \mathbf{y}'_N)\}$
    $f = false$
  end if
until f is $true$

4.3 Convergence

It is easy to see that during each cutting plane iteration, at most two constraints will be added to the constraint sets. Following the ideas in [1, 2], we show that the 2-slack cutting plane algorithm will converge in at most a non-trivial fixed number of iterations by proving the following theorems.

Theorem 4. *In each iteration of Algorithm 3, the value of the dual objective of Equation 5 increases at least*

$$\mu = \frac{1}{2}\min\{\varepsilon_1 C_1, \varepsilon_2 C_2, \frac{\varepsilon_1^2}{4P^2}, \frac{\varepsilon_2^2}{4Q^2}, \frac{(\varepsilon_1+\varepsilon_2)^2}{4P^2+4Q^2+8PQ}\},$$

where

$$P^2 = \max_{i,\mathbf{y}_i\in Y_i,\mathbf{y}'_i\in Y_i} \|\delta\Psi_i(\mathbf{y}_i, \mathbf{y}'_i)\|^2 \tag{6}$$

$$Q^2 = \max_{j,\mathbf{y}_j\in \mathcal{Y}/Y_j,\mathbf{y}'_j\in Y_j} \left\|\delta\Psi_j(\mathbf{y}_j, \mathbf{y}'_j)\right\|^2 . \tag{7}$$

Sketch of the Proof. We prove this theorem by first proving the following lemmas:

Lemma 1: If only one constraint is added to the first constraint set in one iteration, the increment of the dual objective is lower bounded by $\frac{1}{2}\min\{\varepsilon_1 C_1, \varepsilon_1^2/4P^2\}$.
Lemma 2: If only one constraint is added to the second constraint set in one iteration, the increment of the dual objective is lower bounded by $\frac{1}{2}\min\{\varepsilon_2 C_2, \frac{\varepsilon_2^2}{4Q^2}\}$.
Lemma 3: If two constraints are added to the two constraint set respectively, the increment of the dual objective is lower bounded by

$$\frac{1}{2}\min\{(\varepsilon_1+\varepsilon_2)\min\{C_1, C_1\}, \frac{(\varepsilon_1+\varepsilon_2)^2}{4P^2+4Q^2+8PQ}\}.$$

In each iteration, if some constraints are added, the increment of the dual objective is bounded by these three lemmas; if no constraint is added, the algorithm simply halts, and hence we can draw the conclusion that for each cutting plane iteration, the value of the dual objective will be increased by at least

$$\begin{aligned}\mu =&\frac{1}{2}\min\{\varepsilon_1 C_1, \varepsilon_2 C_2, (\varepsilon_1+\varepsilon_2)\min\{C_1, C_2\},\\ &\frac{\varepsilon_1^2}{4P^2}, \frac{\varepsilon_2^2}{4Q^2}, \frac{(\varepsilon_1+\varepsilon_2)^2}{4P^2+4Q^2+8PQ}\} \qquad (8)\\ =&\frac{1}{2}\min\{\varepsilon_1 C_1, \varepsilon_2 C_2, \frac{\varepsilon_1^2}{4P^2}, \frac{\varepsilon_2^2}{4Q^2}, \frac{(\varepsilon_1+\varepsilon_2)^2}{4P^2+4Q^2+8PQ}\} \qquad (9)\end{aligned}$$

The detail of the proof is as follows.

Proof. We assume that the 2 constraints sets are Ω_1 and Ω_2, and $\omega_1 = |\Omega_1|$, $\omega_2 = |\Omega_2|$. The original 2-slack formulation is

$$\min_{\mathbf{w},\xi,\zeta} \frac{1}{2}\|\mathbf{w}\|^2 + C_1\xi + C_2\zeta \qquad (10)$$

$$\forall(\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \Omega_1 \qquad (11)$$

$$\xi \geq \frac{1}{N}\sum_{i=1}^{N}(\Delta(\mathbf{y}_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}_i, \overline{\mathbf{y}_i})\rangle) \qquad (12)$$

$$\forall(\mathbf{y}'_1, \ldots, \mathbf{y}'_N) \in \Omega_2 \qquad (13)$$

$$\zeta \geq \frac{1}{N}\sum_{i=1}^{N}(\Delta(\mathbf{y}'_i, \overline{\mathbf{y}_i}) + \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}'_i, \overline{\mathbf{y}_i})\rangle) \qquad (14)$$

Moreover, we let $(\mathbf{y}_1^{(i)}, \ldots, \mathbf{y}_N^{(i)})$ be the i-th constraint in Ω_1, and $(\mathbf{y}_1^{(j+\omega_1)}, \ldots, \mathbf{y}_N^{(j+\omega_1)})$ be the j-th constraint in Ω_2.

The Lagrangian of the original 2-slack formulation is given by

$$
\begin{aligned}
&L(\mathbf{w},\xi,\zeta,\alpha) = \frac{1}{2}\|\mathbf{w}\|^2 + C_1\xi + C_2\zeta \\
&+ \sum_{i=1}^{\omega_1} \alpha_i [\frac{1}{N}\sum_{j=1}^{N}(\mathit{\Delta}(\mathbf{y}_j^{(i)}, \overline{\mathbf{y}_j}) + \langle \mathbf{w}, \delta\mathit{\Psi}_j(\mathbf{y}_j^{(i)}, \overline{\mathbf{y}_j})\rangle) - \xi] \\
&+ \sum_{i=\omega_1+1}^{\omega_1+\omega_2} \alpha_i [\frac{1}{N}\sum_{j=1}^{N}(\mathit{\Delta}(\mathbf{y}_j^{(i)}, \overline{\mathbf{y}_j}) + \langle \mathbf{w}, \delta\mathit{\Psi}_j(\mathbf{y}_j^{(i)}, \overline{\mathbf{y}_j})\rangle) - \zeta]
\end{aligned} \tag{15}
$$

Differentiating with respect to $\mathbf{w}$ gives

$$
\mathbf{w} = \sum_{i=1}^{\omega_1+\omega_2} \alpha_i \frac{1}{N}\sum_{j=1}^{N} \delta\mathit{\Psi}_j(\overline{\mathbf{y}_j}, \mathbf{y}_j^{(i)}) \tag{16}
$$

Differentiating with respect to ξ and ζ gives

$$
\sum_{i=1}^{\omega_1} \alpha_i = C_1 \tag{17}
$$

$$
\sum_{i=\omega_1+1}^{\omega_1+\omega_2} \alpha_i = C_2 \tag{18}
$$

Plugging $\mathbf{w}$ and constraints on α results in the dual problem:

$$
\max_{\alpha} \sum_{i=1}^{\omega_1+\omega_2} \alpha_i \mathit{\Delta}(i) - \frac{1}{2}\sum_{i=1}^{\omega_1+\omega_2}\sum_{j=1}^{\omega_1+\omega_2} \alpha_i\alpha_j K(i,j) \tag{19}
$$

$$
\sum_{i=1}^{\omega_1} \alpha_i = C_1 \tag{20}
$$

$$
\sum_{i=\omega_1+1}^{\omega_1+\omega_2} \alpha_i = C_2 \tag{21}
$$

where $\mathit{\Delta}(i)$ is defined as $(\frac{1}{N}\sum_{j=1}^{N}\mathit{\Delta}(\mathbf{y}_j^{(i)}, \overline{\mathbf{y}_j}))$, and $K(i,j)$ is the entry on the i-th row, the j-th column of the kernel matrix K defined by

$$
K(i,j) = [\frac{1}{N}\sum_{k=1}^{N}\delta\mathit{\Psi}_k(\overline{\mathbf{y}_k}, \mathbf{y}_k^{(i)})]^T [\frac{1}{N}\sum_{k=1}^{N}\delta\mathit{\Psi}_k(\overline{\mathbf{y}_k}, \mathbf{y}_k^{(j)})] \tag{22}
$$

Lemma 1: Only one constraint is added to $\mathit{\Omega}_1$. Let the constraint be $(\mathbf{y}_1^{(\omega_1+\omega_2+1)}, \ldots, \mathbf{y}_N^{(\omega_1+\omega_2+1)})$.

We let α be the solution of the dual problem before adding this constraint. To lower bound the progress made by the algorithm, we consider the increase in the dual that can be achieved with a line search

$$
\max_{0\leq\beta\leq C_1} \{D(\alpha+\beta\eta)\} - D(\alpha) \tag{23}
$$

where we construct η as:

$$\eta_i = -\frac{1}{C_1}\alpha_i \qquad \textit{for } 1 \leq i \leq \omega_1 \tag{24}$$

$$\eta_i = 0 \qquad \textit{for } \omega_1 + 1 \leq i \leq \omega_1 + \omega_2 \tag{25}$$

$$\eta_i = 1 \qquad \textit{for } i = \omega_1 + \omega_2 + 1 \tag{26}$$

We now need a lower bound for $\nabla D(\alpha)^T \eta$ and an upper bound for $\eta^T K \eta$.

Note that $\frac{\partial D(\alpha)}{\partial \alpha_i} = \Delta(i) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,i) = \xi$ for $\alpha_i \neq 0$, $1 \leq i \leq \omega_1$ and $\frac{\partial D(\alpha)}{\partial \alpha_{\omega_1+\omega_2+1}} = \Delta(\omega_1+\omega_2+1) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j, \omega_1+\omega_2+1) = \xi + \gamma_1 \geq \xi + \varepsilon_1$, indicating that $\nabla D(\alpha)^T \eta = \gamma_1 \geq \varepsilon_1$.

On the other hand, we have

$$\begin{aligned} \eta^T K \eta = K(\omega_1+\omega_2+1, \omega_1+\omega_2+1) - \\ \frac{2}{C_1}\sum_{i=1}^{\omega_1} \alpha_i K(i, \omega_1+\omega_2+1) + \\ \frac{1}{C_1^2}\sum_{i=1}^{\omega_1}\sum_{j=1}^{\omega_1} \alpha_i \alpha_j K(i,j) \end{aligned} \tag{27}$$

$$\leq P^2 + \frac{2}{C_1} C_1 P^2 + \frac{1}{C_1^2} C_1^2 P^2 \tag{28}$$

$$= 4P^2 \tag{29}$$

Thus by using Lemma 2 in [1], the value of the objective will increase at least

$$\max_{0 \leq \beta \leq C_1} \{D(\alpha + \beta\eta)\} - D(\alpha) \geq \frac{1}{2}\min\{\varepsilon_1 C_1, \frac{\varepsilon_1^2}{4P^2}\} \tag{30}$$

Lemma 2: Only one constraint is added to Ω_2. Again let the constraint be $(\mathbf{y}_1^{(\omega_1+\omega_2+1)}, \ldots, \mathbf{y}_N^{(\omega_1+\omega_2+1)})$.

Using the same routine, we consider the increase in the dual that can be achieved with a line search

$$\max_{0 \leq \beta \leq C_2} \{D(\alpha + \beta\eta)\} - D(\alpha) \tag{31}$$

where we construct η as:

$$\eta_i = 0 \qquad \textit{for } 1 \leq i \leq \omega_1 \tag{32}$$

$$\eta_i = -\frac{1}{C_2}\alpha_i \qquad \textit{for } \omega_1 + 1 \leq i \leq \omega_1 + \omega_2 \tag{33}$$

$$\eta_i = 1 \qquad \textit{for } i = \omega_1 + \omega_2 + 1 \tag{34}$$

We now need a lower bound for $\nabla D(\alpha)^T \eta$ and an upper bound for $\eta^T K \eta$.

Note that $\frac{\partial D(\alpha)}{\partial \alpha_i} = \Delta(i) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,i) = \zeta$ for $\alpha_i \neq 0, \omega_1+1 \leq i \leq \omega_1+\omega_2$ and $\frac{\partial D(\alpha)}{\partial \alpha_{\omega_1+\omega_2+1}} = \Delta(\omega_1+\omega_2+1) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,\omega_1+\omega_2+1) = \zeta+\gamma_2 \geq \zeta+\varepsilon_2$, indicating that $\nabla D(\alpha)^T \eta = \gamma_2 \geq \varepsilon_2$.

On the other hand, we have

$$\begin{aligned} \eta^T K \eta = K(\omega_1+\omega_2+1, \omega_1+\omega_2+1) - \\ \frac{2}{C_2} \sum_{i=\omega_1+1}^{\omega_1+\omega_2} \alpha_i K(i, \omega_1+\omega_2+1) + \\ \frac{1}{C_2^2} \sum_{i=\omega_1+1}^{\omega_1+\omega_2} \sum_{j=\omega_1+1}^{\omega_1+\omega_2} \alpha_i \alpha_j K(i,j) \quad (35) \\ \leq Q^2 + \frac{2}{C_2} C_2 Q^2 + \frac{1}{C_2^2} C_2^2 Q^2 \quad (36) \\ = 4Q^2 \quad (37) \end{aligned}$$

Thus by using Lemma 2 in [1], the value of the objective will increase at least

$$\max_{0 \leq \beta \leq C_2} \{D(\alpha+\beta\eta)\} - D(\alpha) \geq \frac{1}{2} \min\{\varepsilon_2 C_2, \frac{\varepsilon_2^2}{4Q^2}\} \quad (38)$$

Lemma 3: One constraint is added to Ω_1 (let it be $(\mathbf{y}_1^{(\omega_1+\omega_2+1)}, \ldots, \mathbf{y}_N^{(\omega_1+\omega_2+1)})$) and one constraint is added to Ω_2 (let it be $(\mathbf{y}_1^{(\omega_1+\omega_2+2)}, \ldots, \mathbf{y}_N^{(\omega_1+\omega_2+2)})$).

We consider the increase in the dual that can be achieved with a line search

$$\max_{0 \leq \beta \leq \min\{C_1, C_2\}} \{D(\alpha+\beta\eta)\} - D(\alpha) \quad (39)$$

where we construct η as:

$$\eta_i = -\frac{1}{C_1}\alpha_i \qquad \textit{for } 1 \leq i \leq \omega_1 \quad (40)$$

$$\eta_i = -\frac{1}{C_2}\alpha_i \qquad \textit{for } \omega_1+1 \leq i \leq \omega_1+\omega_2 \quad (41)$$

$$\eta_i = 1 \qquad \textit{for } i = \omega_1+\omega_2+1, \omega_1+\omega_2+2 \quad (42)$$

Note that $\frac{\partial D(\alpha)}{\partial \alpha_i} = \Delta(i) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,i) = \xi$ for $\alpha_i \neq 0$, $1 \leq i \leq \omega_1$; $\frac{\partial D(\alpha)}{\partial \alpha_i} = \Delta(i) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,i) = \zeta$ for $\alpha_i \neq 0$, $\omega_1+1 \leq i \leq \omega_1+\omega_2$; $\frac{\partial D(\alpha)}{\partial \alpha_{\omega_1+\omega_2+1}} = \Delta(\omega_1+\omega_2+1) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,\omega_1+\omega_2+1) = \xi+\gamma_1 \geq \xi+\varepsilon_1$ $\frac{\partial D(\alpha)}{\partial \alpha_{\omega_1+\omega_2+2}} = \Delta(\omega_1+\omega_2+2) - \sum_{j=1}^{\omega_1+\omega_2} \alpha_j K(j,\omega_1+\omega_2+1) = \zeta+\gamma_2 \geq \zeta+\varepsilon_2$, indicating that $\nabla D(\alpha)^T \eta = \gamma_1+\gamma_2 \geq \varepsilon_1+\varepsilon_2$.

On the other hand, we have

$$
\begin{aligned}
\eta^T K \eta = {} & K(\omega_1+\omega_2+1, \omega_1+\omega_2+1)+ \\
& K(\omega_1+\omega_2+2, \omega_1+\omega_2+2)+ \\
& 2K(\omega_1+\omega_2+1, \omega_1+\omega_2+2)+ \\
& \frac{1}{C_1^2}\sum_{i=1}^{\omega_1}\sum_{j=1}^{\omega_1}\alpha_i\alpha_j K(i,j) - \frac{2}{C_1}\sum_{j=1}^{\omega_1}\alpha_j K(j,\omega_1+\omega_2+1) \\
& -\frac{2}{C_2}\sum_{j=1}^{\omega_1}\alpha_j K(j,\omega_1+\omega_2+2)+ \\
& \frac{1}{C_2}\sum_{i=\omega_1+1}^{\omega_1+\omega_2}\sum_{j=\omega_1+1}^{\omega_1+\omega_2}\alpha_i\alpha_j K(i,j)- \\
& \frac{2}{C_2}\sum_{j=\omega_1+1}^{\omega_1+\omega_2}\alpha_j K(j,\omega_1+\omega_2+1)- \\
& \frac{2}{C_2}\sum_{j=\omega_1+1}^{\omega_1+\omega_2}\alpha_j K(j,\omega_1+\omega_2+2) \\
& +\frac{2}{C_1C_2}\sum_{i=1}^{\omega_1}\sum_{j=\omega_1+1}^{\omega_1+\omega_2}\alpha_i\alpha_j K(i,j) && (43) \\
& \le P^2+Q^2+2PQ+P^2+2P^2+2PQ+ \\
& Q^2+2PQ+2Q^2+2PQ && (44) \\
& = 4P^2+4Q^2+8PQ && (45)
\end{aligned}
$$

Thus by using Lemma 2 in [1], the value of the objective in this case will increase at least

$$
\begin{aligned}
& \max_{0\le\beta\le\min\{C_1,C_2\}}\{D(\alpha+\beta\eta)\}-D(\alpha)\ge \\
& \frac{1}{2}\min\{(\varepsilon_1+\varepsilon_2)\min\{C_1,C_2\}, \frac{(\varepsilon_1+\varepsilon_2)^2}{4P^2+4Q^2+8PQ}\} && (46)
\end{aligned}
$$

Theorem 5. *The value of the dual objective of Equation 5 is upper bounded by $(C_1\Delta_1 + C_2\Delta_2)$, where*

$$\Delta_1 = \max_{i,\mathbf{y}_i\in Y_i,\mathbf{y}'_i\in Y_i}\Delta(\mathbf{y}_i,\mathbf{y}'_i) \quad (47)$$

$$\Delta_2 = \max_{j,\mathbf{y}_j\in Y_i,\mathbf{y}'_j\in \mathcal{Y}/Y_j}\Delta(\mathbf{y}_j,\mathbf{y}'_j) \quad (48)$$

and a feasible starting point of the dual objective could have value 0.

Proof.

$$\max_{\alpha} -\frac{1}{2}\alpha^T K\alpha + \sum_{i=1}^{\omega_1+\omega_2} \alpha_i \Delta(i) \leq \sum_{i=1}^{\omega_1+\omega_2} \alpha_i \Delta(i) \tag{49}$$

$$\leq C_1\Delta_1 + C_2\Delta_2 \tag{50}$$

Buy setting $\Omega_1 = \Omega_2 = \{(\overline{\mathbf{y}_1}, \ldots, \overline{\mathbf{y}_N})\}$(the initialized labels), and the corresponding $\alpha_1 = C_1$, $\alpha_2 = C_2$ and other α_i's be 0, the dual objective value becomes 0. Note that this α is a feasible starting point.

References

1. Joachims, T., Finley, T., Yu, C.: Cutting-plane training of structural svms. Machine Learning pp. 27–59 (2009)
2. Zhang, D., Liu, Y., Si, L., Zhang, J., Lawrence, R.: Multiple instance learning on structured data. In: NIPS, pp. 145–153 (2011)