Dual-Decomposed Learning with Factorwise Oracles for Structured Prediction of Large Output Domain

Xiangru Huang

Joint work \(^1\) with
Ian E.H. Yen\(^\dagger\), Kai Zhong\(^*\), Ruohan Zhang\(^*\), Chia Dai\(^\dagger\),
Pradeep Ravikumar\(^\dagger\) and Inderjit Dhillon\(^*\).

\(^*\) University of Texas at Austin
\(^\dagger\) Carnegie Mellon University

\(^1\)[1] Dual Decomposed Learning with Factorwise Oracle for Structural SVM of Large Output Domain. NIPS 2016.
Outline

Motivations

Key Idea

Methodology Sketch

Experimental Results
Problem Setting

- Classification: learn function \( g : \mathcal{X} \rightarrow \mathcal{Y} \)
Problem Setting

- Classification: learn function $g : \mathcal{X} \rightarrow \mathcal{Y}$
- Structural: Assuming structured dependencies on output $g : \mathcal{X} \rightarrow \mathcal{Y}_1 \times \mathcal{Y}_2 \times \cdots \times \mathcal{Y}_m$
Example: Sequence Labeling

- **Unigram Factor:**
  \[ \theta_u : \mathcal{Y}_t \times \mathcal{X}_t \rightarrow \mathcal{R} \]

- **Bigram Factor:**
  \[ \mathcal{Y}_b = \mathcal{Y}_{t-1} \times \mathcal{Y}_t \]
  \[ \theta_b : \mathcal{Y}_b \rightarrow \mathcal{R} \]

**Figure:** Sequence Labeling
Example: Multi-Label Classification with Pairwise Interaction

▶ Unigram Factor:
\( \theta_u : \mathcal{Y}_k \times \mathcal{X} \rightarrow \mathcal{R} \)

▶ Bigram Factor:
\( \mathcal{Y}_b = \mathcal{Y}_k \times \mathcal{Y}_{k'} \)
\( \theta_b : \mathcal{Y}_b \rightarrow \mathcal{R} \)

\( \mathcal{Y}_u = \{0, 1\} \)
\( \mathcal{Y}_b = \{00, 01, 10, 11\} \)

**Figure:** Multi-Label with Pairwise Interaction
Motivations

- \( g : \mathcal{X} \rightarrow \mathcal{Y}_1 \times \mathcal{Y}_2 \times \cdots \times \mathcal{Y}_m \)
Motivations

- \( g : \mathcal{X} \rightarrow \mathcal{Y}_1 \times \mathcal{Y}_2 \times \cdots \times \mathcal{Y}_m \)
- Learning requires inference per iteration.
- Exact inference is slow: each iteration takes \( O(|\mathcal{Y}_i|^n) \) for n-gram factor, where \( |\mathcal{Y}_i| \geq 3000 \).
Motivations

- \( g : \mathcal{X} \rightarrow \mathcal{Y}_1 \times \mathcal{Y}_2 \times \cdots \times \mathcal{Y}_m \)
- Learning requires inference per iteration.
- Exact inference is slow: each iteration takes \( O(|\mathcal{Y}_i|^n) \) for n-gram factor, where \( |\mathcal{Y}_i| \geq 3000 \).
- Approximation downgrades performance.
Key Idea: Dual Decomposed Learning

- Structural Oracle (joint inference) is too expensive.
Key Idea: Dual Decomposed Learning

- Structural Oracle (joint inference) is too expensive.
- Reduce Structural SVM to Multiclass SVMs via soft enforcement of consistency between factors.
Key Idea: Dual Decomposed Learning

- Structural Oracle (joint inference) is too expensive.
- Reduce Structural SVM to Multiclass SVMs via soft enforcement of consistency between factors.
- (Cheap) Active Sets + Factorwise Oracles + Message Passing (between factors).
Key Idea: Factorwise Oracles

- **Inner-Product (unigram) Factor**: $\theta_w(x, y) = \langle w_y, x \rangle$.
  - Reduces to a primal and dual sparse Extreme Multiclass SVM.
  - Reduce $O(D \cdot |Y_i|)$ to $O(|\mathcal{F}_u| \cdot |A_i|)$ (details see [2]) $^2$.

- **Indicator (bigram) Factor**: $\theta(y_1, y_2) = v_{y_1, y_2}$.
  - Maintain Priority Queue on $v_{y_1, y_2}$.
  - Reduce $O(|Y_1||Y_2|)$ to $O(|A_1||A_2|)$.

---

Methodology Sketch

- Original problem:

\[
\min_w \frac{1}{2} \| w \|_2^2 + C \sum_{i=1}^{n} L(w; x_i, y_i)
\]

\[\text{struct hinge loss}\]

\[\text{Dual-Decomposed into independent problems:}\]

\[
\min_{\alpha} \frac{1}{2} \| \sum_{f \in F} \phi(x_f, y_f) \|^2_2 - \sum_{j \in V} \delta_\alpha_j \]

\[\text{Independent Multiclass SVMs with consistency constraints}\]

\[\alpha_f = \alpha_i, \quad \forall (i, f) \in E.\]

- Standard approach finds feasible descent direction, which however needs joint inference.

---

\(^3\)Simon Julien et al. Block-Coordinate Frank-Wolfe Optimization for Structural SVMs. ICML 2013.
Methodology Sketch

- Original problem:

\[
\min_w \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{n} L(w; x_i, y_i)
\]

\[
\text{struct hinge loss}
\]

- Dual-Decomposed into independent problems:

\[
\min_{\alpha_f \in \Delta \left| \mathcal{Y}_f \right|} G(\alpha) := \frac{1}{2} \sum_{F} \left\| \sum_{f \in F} \phi(x_f, y_f)^T \alpha_f \right\|^2 - \sum_{j \in \mathcal{V}} \delta_j^T \alpha_j
\]

\[
\text{Independent Multiclass SVMs}
\]

with consistency constraints

\[
M_{if} \alpha_f = \alpha_i, \quad \forall (i, f) \in \mathcal{E}.
\]

- Standard approach \(^3\) finds feasible descent direction, which however needs joint inference.

\(^3\)Simon Julien et al. Block-Coordinate Frank-Wolfe Optimization for Structural SVMs. ICML 2013.
Methodology Sketch

- Dual-Decomposed into independent problems:

\[
\min_{\alpha_f \in \Delta^{|Y_f|}} G(\alpha) := \frac{1}{2} \sum_F \left( \sum_{f \in F} \phi(x_f, y_f)^T \alpha_f \right)^2 - \sum_{j \in \mathcal{V}} \delta_j^T \alpha_j
\]

with consistency constraints

\[
M_{j_0} \alpha_f = \alpha_j, \quad \forall (j, f) \in \mathcal{E}
\]

- Augmented Lagrangian Method:

\[
\mathcal{L}(\alpha, \lambda) := \sum_F G_F(\alpha_F) + \frac{\rho}{2} \sum_{(j, f) \in \mathcal{E}} \|M_{j_0} \alpha_f - \alpha_j + \lambda_{j_0}^t\|^2
\]

with incremental updated multipliers

\[
\lambda_{j_0}^{t+1} = \lambda_{j_0}^t + \eta(M_{j_0} \alpha_f^{t+1} - \alpha_j^{t+1})
\]
Methodology Sketch

- Augmented Lagrangian Method:

\[
\mathcal{L}(\alpha, \lambda) := \sum_{F} G_F(\alpha_F) + \frac{\rho}{2} \sum_{(j,f) \in \mathcal{E}} \| M_{jf} \alpha_f - \alpha_j + \lambda^t_{jf} \|^2
\]

with independent multiclass SVMs and messages between factors (sparse)

with incremental updated multipliers

\[
\lambda^{t+1}_{jf} = \lambda^{t}_{jf} + \eta (M_{jf} \alpha^{t+1}_f - \alpha^{t+1}_j)
\]

- Update \( \alpha \) and \( \lambda \) alternatively.
Experiments: Sequence Labeling (on ChineseOCR)

- Chinese OCR: \( N = 12,064, \ T = 14.4, \ D = 400, \ K = 3,039 \).
- \( |\mathcal{Y}_b| = 3,039^2 = 9,235,521 \) (bigram language model).
- Decoding: Viterbi Algorithm.

![Figure: Test Error](image1)

![Figure: Objective](image2)
Experiments: Multi-Label Classification (on RCV1)

- RCV-1: \( N = 23,149, \; D = 47,236 \; , \; K = 228 \).
- \( |\mathcal{F}_b| = 228^2 = 51,984 \) (pairwise interaction).
- Decoding: Linear Program.

![Figure: Test Error](image1.png)
![Figure: Objective](image2.png)