Learning And Mining from DatA

南京大學 NANJING UNIVERSITY

Projection-free Distributed Online Convex Optimization with $O(\sqrt{T})$ Communication Complexity

Yuanyu Wan[1], Wei-Wei Tu[2] and Lijun Zhang[1]

[1]Dept. of Computer Science and Technology, Nanjing University
[2]4Paradigm Inc., Beijing, China

ICML 2020

Outline

Outline

1. Introduction
   - Background
   - The Problem and Our Contributions
2. Our Algorithms
   - D-BOCG for Full Information Setting
   - D-BBCG for Bandit Setting
3. Experiments
4. Conclusion

Outline

1. Introduction
   - Background
   - The Problem and Our Contributions
2. Our Algorithms
   - D-BOCG for Full Information Setting
   - D-BBCG for Bandit Setting
3. Experiments
4. Conclusion

Distributed Online Learning over a Network

- Formal definition

for $t = 1, 2, \ldots, T$ **do**
    for each local learner $i \in [n]$ **do**
        pick a decision $\mathbf{x}_i(t) \in \mathcal{K}$
        receive a convex loss function $f_{t,i}(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$
        communicate with its neighbors and update $\mathbf{x}_i(t)$
    end for
end for

Distributed Online Learning over a Network

Formal definition

1: **for** $t = 1, 2, \ldots, T$ **do**
2:     **for** each local learner $i \in [n]$ **do**
3:         pick a decision $\mathbf{x}_i(t) \in \mathcal{K}$
        receive a convex loss function $f_{t,i}(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$
4:         communicate with its neighbors and update $\mathbf{x}_i(t)$
5:     **end for**
6: **end for**

- the network is defined as $G = (V, E)$, $V = [n]$
- each node $i \in [n]$ is a local learner
- node i can only communicate with its immediate neighbors

$$N_i = \{j \in V | (i, j) \in E\}$$

- the global loss function is defined as $f_t(\mathbf{x}) = \sum_{j=1}^{n} f_{t,j}(\mathbf{x})$

Distributed Online Learning over a Network

- Formal definition

```
for t = 1, 2, ..., T do
    for each local learner i ∈ [n] do
        pick a decision x_i(t) ∈ K
        receive a convex loss function f_{t,i}(x) : K → R
        communicate with its neighbors and update x_i(t)
    end for
end for
```

- Regret of local learner i

$$R_{T,i} = \sum_{t=1}^{T} f_t(\mathbf{x}_i(t)) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

Distributed Online Learning over a Network

- Formal definition

for $t = 1, 2, \ldots, T$ **do**
    for each local learner $i \in [n]$ **do**
        pick a decision $\mathbf{x}_i(t) \in \mathcal{K}$
        receive a convex loss function $f_{t,i}(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$
        communicate with its neighbors and update $\mathbf{x}_i(t)$
    end for
end for

- Regret of local learner i

$$R_{T,i} = \sum_{t=1}^{T} f_t(\mathbf{x}_i(t)) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

- Applications
  - multi-agent coordination
  - distributed tracking in sensor networks

Projection-based Methods

- Distributed Online Dual Averaging [Hosseini et al., 2013]

1: **for** each local learner $i \in [n]$ **do**
2: Play $\mathbf{x}_i(t)$ and compute $\mathbf{g}_i(t) = \nabla f_{t,i}(\mathbf{x}_i(t))$
3: $\mathbf{z}_i(t+1) = \sum_{j \in N_i} P_{ij}\mathbf{z}_j(t) + \mathbf{g}_i(t)$
4: $\mathbf{x}_i(t+1) = \Pi_{\mathcal{K}}^{\psi}(\mathbf{z}_i(t+1), \alpha(t))$
5: **end for**

- $P_{ij} > 0$ only if $(i,j) \in E$ or $P_{ij} = 0$
- $\psi(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ is a proximal function, e.g., $\psi(\mathbf{x}) = \|\mathbf{x}\|_2^2$
- projection step: $\Pi_{\mathcal{K}}^{\psi}(\mathbf{z}, \alpha) = \operatorname{argmin}_{\mathbf{x} \in \mathcal{K}} \mathbf{z}^\top \mathbf{x} + \frac{1}{\alpha}\psi(\mathbf{x})$
- $\alpha(t) = O(1/\sqrt{t}) \to R_{T,i} = O(\sqrt{T})$

Projection-based Methods

- Distributed Online Dual Averaging [Hosseini et al., 2013]

1: **for** each local learner $i \in [n]$ **do**
2: Play $\mathbf{x}_i(t)$ and compute $\mathbf{g}_i(t) = \nabla f_{t,i}(\mathbf{x}_i(t))$
3: $\mathbf{z}_i(t+1) = \sum_{j \in N_i} P_{ij}\mathbf{z}_j(t) + \mathbf{g}_i(t)$
4: $\mathbf{x}_i(t+1) = \Pi_{\mathcal{K}}^{\psi}(\mathbf{z}_i(t+1), \alpha(t))$
5: **end for**

  - $P_{ij} > 0$ only if $(i,j) \in E$ or $P_{ij} = 0$
  - $\psi(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ is a proximal function, e.g., $\psi(\mathbf{x}) = \|\mathbf{x}\|_2^2$
  - projection step: $\Pi_{\mathcal{K}}^{\psi}(\mathbf{z}, \alpha) = \operatorname{argmin}_{\mathbf{x} \in \mathcal{K}} \mathbf{z}^\top \mathbf{x} + \frac{1}{\alpha}\psi(\mathbf{x})$
  - $\alpha(t) = O(1/\sqrt{t}) \to R_{T,i} = O(\sqrt{T})$

- Distributed Online Gradient Descent [Ram et al., 2010]
  - also need a projection step

Projection-free Methods

- Motivation: the projection step could be time-consuming
  - if $\mathcal{K}$ is a trace norm ball, it requires SVD of a matrix

Projection-free Methods

- Motivation: the projection step could be time-consuming
  - if $\mathcal{K}$ is a trace norm ball, it requires SVD of a matrix
- Distributed Online Conditional Gradient [Zhang et al., 2017]

1: **for** each local learner $i \in [n]$ **do**
2: Play $\mathbf{x}_i(t)$ and compute $\mathbf{g}_i(t) = \nabla f_{t,i}(\mathbf{x}_i(t))$
3: $\mathbf{v}_i = \text{argmin}_{\mathbf{x} \in \mathcal{K}} \nabla F_{t,i}(\mathbf{x}_i(t))^\top \mathbf{x}$
4: $\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + s_t(\mathbf{v}_i - \mathbf{x}_i(t))$
5: $\mathbf{z}_i(t+1) = \sum_{j \in N_i} P_{ij} \mathbf{z}_j(t) + \mathbf{g}_i(t)$
6: **end for**

- $F_{t,i}(\mathbf{x}) = \eta \mathbf{z}_i(t)^\top \mathbf{x} + \|\mathbf{x} - \mathbf{x}_1(1)\|_2^2$
- $\eta = O(T^{-3/4}), s_t = 1/\sqrt{t} \to R_{T,i} = O(T^{3/4})$
- only contain linear optimization step (step 3)
- T communication rounds

Outline

Question

Can the $O(T)$ communication complexity of distributed online conditional gradient (D-OCG) be reduced?

Question

Can the $O(T)$ communication complexity of distributed online conditional gradient (D-OCG) be reduced?

- An affirmative and non-trivial answer
  - distributed block online conditional gradient (D-BOCG)
  - communication complexity: from $O(T)$ to $O(\sqrt{T})$
  - regret bound: $O(T^{3/4})$

Question

Can the $O(T)$ communication complexity of distributed online conditional gradient (D-OCG) be reduced?

- An affirmative and non-trivial answer
  - distributed block online conditional gradient (D-BOCG)
  - communication complexity: from $O(T)$ to $O(\sqrt{T})$
  - regret bound: $O(T^{3/4})$
- An extension to the bandit setting
  - distributed block bandit conditional gradient (D-BBCG)
  - communication complexity: $O(\sqrt{T})$
  - high-probability regret bound: $\widetilde{O}(T^{3/4})$

Outline

LAMDA Learning And Mining from DatA

Outline

LAMDA Learning And Mining from DatA

Main Idea

- Delayed update mechanism

  - only update in the beginning of each block
  - only need $\sqrt{T}$ communication rounds

Main Idea

- Delayed update mechanism

  - only update in the beginning of each block
  - only need $\sqrt{T}$ communication rounds
- Iterative linear optimization steps
  - recall the update rules of D-OCG

$$\mathbf{v}_i = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \nabla F_{t,i}(\mathbf{x}_i(t))^\top \mathbf{x}$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + s_t(\mathbf{v}_i - \mathbf{x}_i(t))$$

  - delayed update + D-OCG: a worse regret bound
  - multiple linear optimization steps for each update

Conditional Gradient with Stopping Condition (CGSC)

- CGSC [Garber and Kretzu, 2019]

1: **Input**: feasible set $\mathcal{K}$, $\epsilon > 0$, L, $F(\mathbf{x})$, $\mathbf{x}_{\text{in}}$
2: $\tau = 0, \mathbf{c}_1 = \mathbf{x}_{\text{in}}$
3: **repeat**
4: $\quad \tau = \tau + 1$
5: $\quad \mathbf{v}_\tau \in \operatorname*{argmin}_{\mathbf{x} \in \mathcal{K}} \nabla F(\mathbf{c}_\tau)^\top \mathbf{x}$
6: $\quad s_\tau = \operatorname*{argmin}_{s \in [0,1]} F(\mathbf{c}_\tau + s(\mathbf{v}_\tau - \mathbf{c}_\tau))$
7: $\quad \mathbf{c}_{\tau+1} = \mathbf{c}_\tau + s_\tau(\mathbf{v}_\tau - \mathbf{c}_\tau)$
8: **until** $\nabla F(\mathbf{c}_\tau)^\top(\mathbf{c}_\tau - \mathbf{v}_\tau) \leq \epsilon$ or $\tau = L$
9: **return** $\mathbf{x}_{\text{out}} = \mathbf{c}_\tau$

  - $F(\mathbf{x}_{\text{out}})$ is very small with appropriate L and ϵ
  - it was widely studied [Frank and Wolfe, 1956, Jaggi, 2013]

The Proposed D-BOCG Algorithm

Initialization: choose $\{\mathbf{x}_i(1) = \mathbf{0} \in \mathcal{K} | i \in V\}$ and set $\{\mathbf{z}_i(1) = \mathbf{0} | i \in V\}$

for $t = 1, \cdots, T$ **do**

  $m_t = \lceil t/K \rceil$

  for each local learner $i \in V$ **do**

    if $t > 1$ and $\mathrm{mod}(t, K) = 1$ **then**

      $\widehat{\mathbf{g}}_i(m_t - 1) = \sum_{k=t-K}^{t-1} \mathbf{g}_i(k)$

      $\mathbf{z}_i(m_t) = \sum_{j \in N_i} P_{ij}\mathbf{z}_j(m_t - 1) + \widehat{\mathbf{g}}_i(m_t - 1)$

      define $F_{m_t,i}(\mathbf{x}) = \eta \mathbf{z}_i(m_t)^\top \mathbf{x} + \|\mathbf{x}\|_2^2$

      $\mathbf{x}_i(m_t) = \mathrm{CGSC}(\mathcal{K}, \epsilon, L, F_{m_t,i}(\mathbf{x}), \mathbf{x}_i(m_t - 1))$

    end if

    play $\mathbf{x}_i(m_t)$ and observe $\mathbf{g}_i(t) = \nabla f_{t,i}(\mathbf{x}_i(m_t))$

  end for

end for

Learning And Mining from DatA

Regret of D-BOCG

Theorem 1

Let $\eta = O(T^{-3/4})$, $\epsilon = O(T^{-1/2})$, $K = \sqrt{T}$ *and* $L = O(\sqrt{T})$*. For any* $i \in V$*, D-BOCG has*

$$R_{T,i} \leq O(GRT^{3/4}).$$

- Assumptions
  - $|f_{t,i}(\mathbf{x}) - f_{t,i}(\mathbf{y})| \leq G\|\mathbf{x} - \mathbf{y}\|_2$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$
  - $r\mathcal{B}^d \subseteq \mathcal{K} \subseteq R\mathcal{B}^d$, $\mathcal{B}^d$ is the unit Euclidean ball
  - $P \in \mathbb{R}^{n \times n}$ is symmetric and doubly stochastic, i.e.,

$$P = P^\top, \mathbf{1}^\top P = \mathbf{1}^\top, P\mathbf{1} = \mathbf{1}$$

Regret of D-BOCG

Theorem 1

Let $\eta = O(T^{-3/4})$, $\epsilon = O(T^{-1/2})$, $K = \sqrt{T}$ and $L = O(\sqrt{T})$. For any $i \in V$, D-BOCG has

$$R_{T,i} \le O(GRT^{3/4}).$$

- Assumptions
  - $|f_{t,i}(\mathbf{x}) - f_{t,i}(\mathbf{y})| \le G\|\mathbf{x} - \mathbf{y}\|_2$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$
  - $r\mathcal{B}^d \subseteq \mathcal{K} \subseteq R\mathcal{B}^d$, $\mathcal{B}^d$ is the unit Euclidean ball
  - $P \in \mathbb{R}^{n \times n}$ is symmetric and doubly stochastic, i.e.,

$$P = P^\top, \mathbf{1}^\top P = \mathbf{1}^\top, P\mathbf{1} = \mathbf{1}$$

- Remarks
  - regret bound: $R_{T,i} = O(T^{3/4})$
  - #communication rounds: $T/K = \sqrt{T}$
  - #linear optimization steps: $LT/K = O(T)$

Outline

Standard Technique

- Bandit setting
  - only the loss value is available to learners
  - the main challenge is due to the lack of gradient

Standard Technique

- Bandit setting
  - only the loss value is available to learners
  - the main challenge is due to the lack of gradient
- One-point Gradient Estimator [Flaxman et al., 2005]
  - δ-smoothed version of $f(\mathbf{x})$

$$\widehat{f}_\delta(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{B}^d}[f(\mathbf{x}+\delta\mathbf{u})]$$

  - let $\delta > 0$ and $\mathcal{S}^d$ be the unit sphere

$$\nabla\widehat{f}_\delta(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{S}^d}\left[\frac{d}{\delta}f(\mathbf{x}+\delta\mathbf{u})\mathbf{u}\right]$$

  - only observe the single value $f(\mathbf{x}+\delta\mathbf{u})$

Standard Technique

- Bandit setting
  - only the loss value is available to learners
  - the main challenge is due to the lack of gradient
- One-point Gradient Estimator [Flaxman et al., 2005]
  - δ-smoothed version of $f(\mathbf{x})$

$$\widehat{f}_\delta(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{B}^d}[f(\mathbf{x}+\delta\mathbf{u})]$$

  - let $\delta > 0$ and $\mathcal{S}^d$ be the unit sphere

$$\nabla\widehat{f}_\delta(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{S}^d}\left[\frac{d}{\delta}f(\mathbf{x}+\delta\mathbf{u})\mathbf{u}\right]$$

  - only observe the single value $f(\mathbf{x}+\delta\mathbf{u})$
- A smaller set $\mathcal{K}_\delta \subseteq \mathcal{K}$
  - $\mathcal{K}_\delta = (1-\delta/r)\mathcal{K} = \{(1-\delta/r)\mathbf{x}|\mathbf{x}\in\mathcal{K}\}, 0 < \delta \leq r$
  - $\mathbf{x}+\delta\mathbf{u}\in\mathcal{K}$ for $\mathbf{x}\in\mathcal{K}_\delta, \mathbf{u}\sim\mathcal{S}$

The Proposed D-BBCG Algorithm

Initialization: choose $\{\mathbf{x}_i(1) = \mathbf{0} \in \mathcal{K}_\delta | i \in V\}$ and set $\{\mathbf{z}_i(1) = \mathbf{0} | i \in V\}$
for $t = 1, \cdots, T$ **do**
  $m_t = \lceil t/K \rceil$
  for each local learner $i \in V$ **do**
    if $t > 1$ and $\mathrm{mod}(t, K) = 1$ **then**
      $\widehat{\mathbf{g}}_i(m_t - 1) = \sum_{k=t-K}^{t-1} \mathbf{g}_i(k)$
      $\mathbf{z}_i(m_t) = \sum_{j \in N_i} P_{ij} \mathbf{z}_j(m_t - 1) + \widehat{\mathbf{g}}_i(m_t - 1)$
      define $F_{m_t,i}(\mathbf{x}) = \eta \mathbf{z}_i(m_t)^\top \mathbf{x} + \|\mathbf{x}\|_2^2$
      $\mathbf{x}_i(m_t) = \mathrm{CGSC}(\mathcal{K}_\delta, \epsilon, L, F_{m_t,i}(\mathbf{x}), \mathbf{x}_i(m_t - 1))$
    end if
    $\mathbf{u}_i(t) \sim \mathcal{S}^d$
    play $\mathbf{y}_i(t) = \mathbf{x}_i(m_t) + \delta \mathbf{u}_i(t)$ and observe $f_{t,i}(\mathbf{y}_i(t))$
    $\mathbf{g}_i(t) = \frac{d}{\delta} f_{t,i}(\mathbf{y}_i(t)) \mathbf{u}_i(t)$
  end for
end for

Regret of D-BBCG

Theorem 2

Let $\eta = O(T^{-3/4})$, $\delta = O(T^{-1/4})$, $\epsilon = O(T^{-1/2})$, $K = T^{1/2}$ and $L = O(\sqrt{T})$. For any $i \in V$, with high probability, D-BBCG has

$$R_{T,i} \leq \tilde{O}(T^{3/4}).$$

- Additional Assumption
  - all local loss functions are chosen beforehand

Regret of D-BBCG

Theorem 2

Let $\eta = O(T^{-3/4})$, $\delta = O(T^{-1/4})$, $\epsilon = O(T^{-1/2})$, $K = T^{1/2}$ and $L = O(\sqrt{T})$. For any $i \in V$, with high probability, D-BBCG has

$$R_{T,i} \leq \widetilde{O}(T^{3/4}).$$

- Additional Assumption
  - all local loss functions are chosen beforehand
- Remarks
  - high-probability regret bound: $R_{T,i} = \widetilde{O}(T^{3/4})$
  - #communication rounds: $T/K = \sqrt{T}$
  - #linear optimization steps: $LT/K = O(T)$

Outline

Experimental Settings

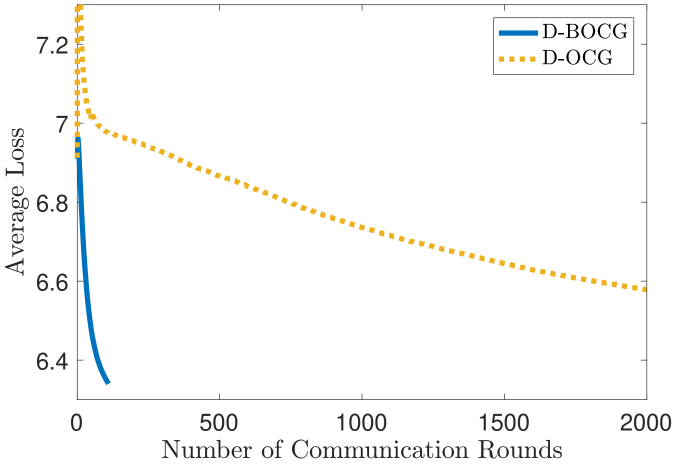
- Distributed multiclass classification [Zhang et al., 2017]

1: **for** $t = 1, 2, \ldots, T$ **do**
2: **for** each local learner $i \in [n]$ **do**
3: receive an example $\mathbf{e}_i(t) \in \mathbb{R}^k$, and choose $X_i(t) = [\mathbf{x}_1^\top; \mathbf{x}_2^\top; \cdots; \mathbf{x}_h^\top] \in \mathcal{K}$
4: receive the true label $y_i(t)$, and suffer the multivariate logistic loss $f_{t,i}(X_i(t)) = \log\left(1 + \sum_{\ell \neq y_i(t)} e^{\mathbf{x}_\ell^\top \mathbf{e}_i(t) - \mathbf{x}_{y_i(t)}^\top \mathbf{e}_i(t)}\right)$
5: communicate with its neighbors and update $X_i(t)$
6: **end for**
7: **end for**

- $\mathcal{K} = \{X \in \mathbb{R}^{h \times k} | \|X\|_* \leq \tau\}$, where $\|X\|_*$ denotes the trace norm of X and τ is a constant
- the network is a cycle graph with 9 nodes

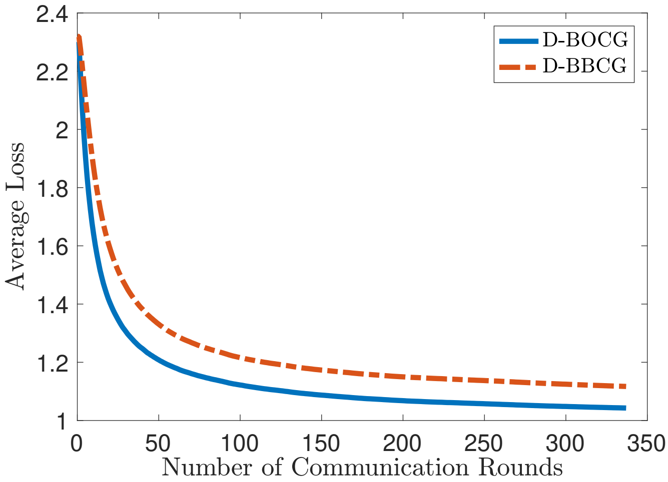
Experimental Results

- aloi dataset from the LIBSVM repository [Chang and Lin, 2011]



Experimental Results

- poker dataset from the LIBSVM repository



Outline

Conclusion and Future Work

- Conclusion
  - D-BOCG enjoys an $O(T^{3/4})$ regret bound with only $O(\sqrt{T})$ communication rounds
  - D-BBCG for bandit setting enjoys a high-probability $\widetilde{O}(T^{3/4})$ regret bound with only $O(\sqrt{T})$ communication rounds
- Future Work
  - improve the regret bound of projection-free distributed online learning by utilizing the curvature of functions

Reference I

Thanks!

Chang, C.-C. and Lin, C.-J. (2011).
LIBSVM: A library for support vector machines.
ACM Transactions on Intelligent Systems and Technology, 2(27):1–27.

Flaxman, A. D., Kalai, A. T., and McMahan, H. B. (2005).
Online convex optimization in the bandit setting: Gradient descent without a gradient.
In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 385–394.

Frank, M. and Wolfe, P. (1956).
An algorithm for quadratic programming.
Naval Research Logistics Quarterly, 3(1–2):95–110.

Garber, D. and Kretzu, B. (2019).
Improved regret bounds for projection-free bandit convex optimization.
arXiv:1910.03374.

Hosseini, S., Chapman, A., and Mesbahi, M. (2013).
Online distributed optimization via dual averaging.
In *52nd IEEE Conference on Decision and Control*, pages 1484–1489.

Jaggi, M. (2013).
Revisiting frank-wolfe: Projection-free sparse convex optimization.
In *Proceedings of the 30th International Conference on Machine Learning*, pages 427–435.

LAMDA
Learning And Mining from DatA

Reference II

Ram, S. S., Nedić, A., and Veeravalli, V. V. (2010).
Distributed stochastic subgradient projection algorithms for convex optimization.
Journal of Optimization Theory and Applications, 147(3):516–545.

Zhang, W., Zhao, P., Zhu, W., Hoi, S. C. H., and Zhang, T. (2017).
Projection-free distributed online learning in networks.
In *Proceedings of the 34th International Conference on Machine Learning*, pages 4054–4062.