FastXML: A Fast and Stable Tree-classifier for eXtreme Multi-label Learning

Yashoteja Prabhu
Indian Institute of Technology Delhi

Manik Varma
Microsoft Research
Extreme Multi-Label Learning

• Learning with millions of categories
• New paradigm for ranking and recommendation

\[ f : X \rightarrow 2^Y \]

X: Users

Y: Items
Extreme Multi-Label Learning

- Learning with millions of categories
- New paradigm for ranking and recommendation
Ranking and Recommendation

- Traditional approaches

\[ h : (X, Y) \rightarrow \{ \times, \checkmark \} \]

\[ h(\text{man}, \text{mango}) \rightarrow \checkmark \]

\[ h(\text{man}, \text{apple}) \rightarrow \times \]
Ranking and Recommendation

- Traditional approaches
Applications – Query Recommendation

- Multi Label Random Forest (MLRF)
- Agrawal, Gupta, Prabhu, Varma – WWW’13
Applications – Video Recommendation

- Label Partitioning for Sublinear Ranking (LPSR)
  - Weston, Makadia, Yee – ICML’13
MLRF and LPSR – Advantages

• Logarithmic time prediction in milliseconds

• Scale to more than a million labels

• Automatically harvest training data

• Handle data set biases including missing and noisy labels

• Significantly outperform traditional techniques
MLRF and LPSR – Shortcomings

• Require expensive training for accurate prediction

• Can learn imbalanced trees

• Room for further improvement in prediction accuracy
FastXML

• We propose a novel rank-sensitive node partitioning objective function based on nDCG

• The proposed node partitioning function
  • Can be optimized more efficiently than MLRF and LPSR via alternating minimization
  • Learns more balanced trees
  • Leads to higher prediction accuracies

• Train on problems with $10^5 - 10^6$ labels in 9 hours using 1 core and show accuracy gains of 5 – 20%
Tree Based Extreme Classification

• Prediction in logarithmic time
FastXML Overview
Node Partitioning in Feature Space
Node Partitioning in Feature Space
Node Partitioning – Initialization

\[ \delta_i \sim \text{Bernoulli}(0.5), \forall i \]
Node Partitioning – Label Ranking

\[ r^{\pm*} = \text{rank} \left( \sum_{i: \delta_i = \pm 1} N_{y_i y_i} \right) \]
Node Partitioning – Label Ranking

$$r^{\pm*} = \text{rank} \left( \sum_{i: \delta_i = \pm 1} N_{y_i y_i} \right)$$
Node Partitioning — Instance Assignment

\[ \delta_i^* = \text{sign}(v_i^- - v_i^+), \forall i \]

Where,

\[ v_i^\pm = -C_r \sum_j \frac{N_{y_i y_{ij}}}{\log(1 + r_j^\pm)} \]
Node Partitioning — Instance Assignment

\[ \delta_i^* = \text{sign}(v_i^- - v_i^+), \forall i \]

Where,

\[ v_i^\pm = -C_r \sum_j \frac{N_{yi}y_{ij}}{\log(1 + r_j^\pm)} \]
Node Partitioning – Instance Assignment

\[ \delta_i^* = \text{sign}(v_i^+ - v_i^-), \forall i \]

Where,

\[ v_i^\pm = -C_r \sum_j \frac{N_{yi}y_{ij}}{\log(1 + r_j^\pm)} \]
Node Partitioning – Label Ranking

\[ r^{\pm*} = \text{rank} \left( \sum_{i: \delta_i = \pm 1} N_{y_i y_i} \right) \]
Node Partitioning – Instance Assignment

\[ \delta_i^* = \text{sign}(v_i^- - v_i^+), \forall i \]

Where,

\[ v_i^\pm = -C_r \sum_j \frac{N_{y_i}y_{ij}}{\log(1 + r_j^\pm)} \]
Node Partitioning – Feature Space Partition

\[ \text{Min}_w \quad \|w\|_1 \]
\[ + \sum_i C_\delta(\delta_i) \log(1 + e^{-\delta_i w^T x_i}) \]
Node Partitioning

\[ \min_{w, \delta, r^\pm} \|w\|_1 + \sum_i C_\delta(\delta_i) \log \left( 1 + e^{-\delta_i w^t x_i} \right) \]

\[ - C_r \left[ \frac{1}{\log(1 + r_1^\pm)} \cdots \frac{1}{\log(1 + r_j^\pm)} \cdots \frac{1}{\log(1 + r_L^\pm)} \right] \left( \sum_{i: \delta_i = \pm 1} N_{y_i} y_i \right) \]
\[
\text{Min}_{w, \delta, r^\pm} \|w\|_1 + \sum_i C_\delta(\delta_i) \log(1 + e^{-\delta_i w^t x_i}) - C_r \left[ \frac{1}{\log(1 + r_1^\pm)} \cdots \frac{1}{\log(1 + r_j^\pm)} \cdots \frac{1}{\log(1 + r_L^\pm)} \right] \left( \sum_{i: \delta_i = \pm 1} N_{y_i} y_i \right)
\]
Node Partitioning

$$\min_{w, \delta, r^\pm} \|w\|_1 + \sum_i C_\delta(\delta_i) \log \left( 1 + e^{-\delta_i w^t x_i} \right)$$

$$- C_r \begin{bmatrix} \frac{1}{\log(1 + r_1^\pm)} & \cdots & \frac{1}{\log(1 + r_j^\pm)} & \cdots & \frac{1}{\log(1 + r_L^\pm)} \end{bmatrix} \left( \sum_{i: \delta_i = \pm 1} N_{y_i} y_i \right)$$
Node Partitioning

\[
\begin{align*}
\min_{w, \delta, r^\pm} \quad & \|w\|_1 + \sum_i C_\delta(\delta_i) \log(1 + e^{-\delta_i w^t x_i}) \\
- \quad & C_r \left[ \frac{1}{\log(1 + r_1^\pm)} \cdots \frac{1}{\log(1 + r_j^\pm)} \cdots \frac{1}{\log(1 + r_L^\pm)} \right] \left( \sum_{i: \delta_i = \pm 1} N_{y_i, y_i} \right)
\end{align*}
\]
Node Partitioning

$$\min_{w, \delta, r^\pm} \|w\|_1 + \sum_i c_\delta(\delta_i) \log(1 + e^{-\delta_i w^t x_i})$$

$$- c_r \left[ \frac{1}{\log(1 + r_1^\pm)} \ldots \frac{1}{\log(1 + r_j^\pm)} \ldots \frac{1}{\log(1 + r_L^\pm)} \right] \left( \sum_{i: \delta_i = \pm 1} n_{y_i} y_i \right)$$

[Diagram of a tree-like structure with fruit icons, representing node partitioning.]
## Data Set Statistics

### Large data sets

<table>
<thead>
<tr>
<th>Data Set</th>
<th># of Training Points (M)</th>
<th># of Test Points (M)</th>
<th># of Dimensions (M)</th>
<th># of Labels (M)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WikiLSHTC</td>
<td>1.89</td>
<td>0.47</td>
<td>1.62</td>
<td>0.33</td>
</tr>
<tr>
<td>Ads-430K</td>
<td>1.12</td>
<td>0.50</td>
<td>0.088</td>
<td>0.43</td>
</tr>
<tr>
<td>Ads-1M</td>
<td>3.92</td>
<td>1.56</td>
<td>0.16</td>
<td>1.08</td>
</tr>
</tbody>
</table>

### Small data sets

<table>
<thead>
<tr>
<th>Data Set</th>
<th># of Training Points</th>
<th># of Test Points</th>
<th># of Dimensions</th>
<th># of Labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delicious</td>
<td>12,920</td>
<td>3,185</td>
<td>500</td>
<td>983</td>
</tr>
<tr>
<td>MediaMill</td>
<td>30,993</td>
<td>12,914</td>
<td>120</td>
<td>101</td>
</tr>
<tr>
<td>RCV1-X</td>
<td>781,265</td>
<td>23,149</td>
<td>47,236</td>
<td>2,456</td>
</tr>
<tr>
<td>BibTeX</td>
<td>4,880</td>
<td>2,515</td>
<td>1,836</td>
<td>159</td>
</tr>
</tbody>
</table>
Results on Small Data Sets

**Delicious**

- FastXML
- MLRF
- LPSR
- 1-vs-All

**MediaMill**

- FastXML
- MLRF
- LPSR
- 1-vs-All

**RCV1-X**

- FastXML
- MLRF
- LPSR
- 1-vs-All

**BibTeX**

- FastXML-T
- FastXML
- MLRF
- LPSR
- 1-vs-All
Large Data Sets - WikiLSHTC

### Dataset Statistics

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Points</td>
<td>1,892,600</td>
</tr>
<tr>
<td>Features</td>
<td>1,617,899 (sparse)</td>
</tr>
<tr>
<td>Labels</td>
<td>325,056</td>
</tr>
<tr>
<td>Test Points</td>
<td>472,835</td>
</tr>
</tbody>
</table>

### Precision at K

<table>
<thead>
<tr>
<th>K</th>
<th>FastXML</th>
<th>LPSR-NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>50</td>
<td>25</td>
</tr>
<tr>
<td>P3</td>
<td>30</td>
<td>15</td>
</tr>
<tr>
<td>P5</td>
<td>20</td>
<td>5</td>
</tr>
</tbody>
</table>

### Training Time (hr)

<table>
<thead>
<tr>
<th></th>
<th>FastXML</th>
<th>LPSR-NB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

### Test Time (min)

<table>
<thead>
<tr>
<th></th>
<th>FastXML</th>
<th>LPSR-NB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
Large Data Sets – Ads-430K

Dataset Statistics

<table>
<thead>
<tr>
<th>Training Points</th>
<th>1,118,084</th>
</tr>
</thead>
<tbody>
<tr>
<td>Features</td>
<td>87,890 (sparse)</td>
</tr>
<tr>
<td>Labels</td>
<td>434,594</td>
</tr>
<tr>
<td>Test Points</td>
<td>502,926</td>
</tr>
</tbody>
</table>

Precision at K

<table>
<thead>
<tr>
<th></th>
<th>FastXML</th>
<th>LPSR-NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Training Time (hr)

Test Time (min)
Large Data Sets – Ads-1M

### Dataset Statistics

<table>
<thead>
<tr>
<th>Category</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Points</td>
<td>3,917,928</td>
</tr>
<tr>
<td>Features</td>
<td>164,592 (sparse)</td>
</tr>
<tr>
<td>Labels</td>
<td>1,082,898</td>
</tr>
<tr>
<td>Test Points</td>
<td>1,563,137</td>
</tr>
</tbody>
</table>

### Training Time (hr)

- FastXML: 5
- LPSR-NB: 3

### Test Time (min)

- FastXML: 5
- LPSR-NB: 8

### Precision at K

- P1: FastXML > LPSR-NB
- P3: FastXML > LPSR-NB
- P5: FastXML > LPSR-NB
Training Time & Speedup vs. Cores

Training Time (hr) vs. Cores

Speedup vs. Cores
Multiple Iterations - Ads-430K

- **w update Iterations**
  - Bars for each iteration level (1 to 5) across different categories.

- **Obj. Value at Root Node**
  - Bars showing obj. value across different categories.

- **Training Time (hr)**
  - Bars representing training time in hours for different categories.

- **Precision at K**
  - Bars indicating precision at K for different categories (P1, P3, P5).
Tree Balance

**Small Data Sets**
- Delicious
- MediaMill
- RCV1-X
- BibTeX

**Large Data Sets**
- WikiLSHTC
- Ads-430K
- Ads-1M

Graphs show the comparison of FastXML, MLRF, and LPSR in both small and large data sets.
Variants of FastXML - Small Data Sets

Delicious

- FastXML
- MLRF-nDCG
- FastXML-nDCG5
- FastXML-P5

RCV1-X

- FastXML
- MLRF-nDCG
- FastXML-nDCG5
- FastXML-P5

MediaMill

- FastXML
- MLRF-nDCG
- FastXML-nDCG5
- FastXML-P5

BibTeX

- FastXML
- MLRF-nDCG
- FastXML-nDCG5
- FastXML-P5
Variants of FastXML - Large Data Sets

WikiLSHTC

Ads-430K

Ads-1M
Conclusions

• FastXML can improve prediction accuracy by as much as 20% over the state-of-the-art

• FastXML can train on problems with a million categories in a few hours on a single desktop