In several machine learning applications, the label space can be enormous, containing even millions of different classes. To handle this, one can organize labels into a tree as in multi-label softmax (HSM) [4], several very popular tools, such as fastText [1] and Learned Trees [6], apply the pick-one-label heuristic, which does not lead to a consistent solution. Probabilistic label trees are a non-regret generalization of HSM to XLMC (XMLC under precision@k).

### Hierarchical softmax

HSM [4] is a multi-class classification algorithm based on a label tree. Each label \( y \) coded by \( z = (z_1, \ldots, z_m) \in C \), where \( C \) is the set of all labels. An internal node identified by \( z \) has its children labeled \( z_i \), \( i = 1, \ldots, m \). The code does not have to be binary.

- **Marginal probability** of a label \( \psi(y) = \prod_i \psi(y_i) \).

- For a multi-class distribution \( \sum_y \psi(y) = 1 \).

- **Regret** of a classifier \( \hat{y} \) with respect to \( \psi(y) \) is \( \sum_y \psi(y) \max_i \{ 1 - \hat{y}_i \} \).

- The regret of a classifier \( \hat{y} \) is minimizing the expected loss:
  \[
  \mathcal{L}_\psi(\hat{y}) = \mathcal{L}_{\psi}(y) - \mathcal{L}_{\psi}(\hat{y}) = \sum_y \psi(y) \max_i \{ 1 - \hat{y}_i \}.
  \]

- **Precision@k** is defined as:
  \[
  \text{Precision@k}(y, \hat{y}, k) = \frac{1}{k} \sum_{i=1}^{k} I(\hat{y}_i = y_i)
  \]

### Probabilistic label trees

PLT [3] are a non-regret generalization of HSM to multi-label problems.

- **Extended code** \( z = (z_1, \ldots, z_h) \).

- **Factorization** of the marginal probability:
  \[
  \psi(y) = \prod_{h=1}^{H} \psi(h, y) \cdot \prod_{h=1}^{H} \psi(h, y).
  \]

- **Multi-label data: Pick-one-label heuristic**
  - Tools like fastText [1] or Learned Trees [6], apply a pick-one-label heuristic to HSM to transform multi-label instances to multi-class ones.
  - Randomly picking a positive label transforms the multi-label distribution to a multi-class distribution:
    \[
    \psi(y) = \sum_{h=1}^{H} \psi(h, y) \cdot \prod_{i=1}^{m} \psi(y_i) = \prod_{i=1}^{m} \psi(y_i).
    \]

- **Inconsistent** (non-zero regret) for label-wise logistic loss and precision@k.

- **Implementation (extremeText)**

  - **Based on fastText**
    - Tree structure: random, Huffman tree or build via top-down hierarchical balanced clustering
    - Linear models in the nodes
    - Online training with features embedding (hidden, dense representation)

  - **12 regularization for all parameters of the model (for embedding and internal node classifiers).**

  - **Hidden representation obtained by weighted average of the feature vectors of proportion to the target score.**

  - **Depth first search prediction for fast online prediction.**

Source code: [https://github.com/mwydmuch/extremeText](https://github.com/mwydmuch/extremeText)

### Experimental results

<table>
<thead>
<tr>
<th></th>
<th>R10@5</th>
<th>R10@3</th>
<th>R10@1</th>
<th>U10@5</th>
<th>U10@3</th>
<th>U10@1</th>
</tr>
</thead>
<tbody>
<tr>
<td>WALLHITCH</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
<tr>
<td>YAHOO</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
<tr>
<td>PROBABILITY</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
</tbody>
</table>
| Ablation analysis for Amazon-670K

<table>
<thead>
<tr>
<th></th>
<th>R10@5</th>
<th>R10@3</th>
<th>R10@1</th>
<th>U10@5</th>
<th>U10@3</th>
<th>U10@1</th>
</tr>
</thead>
<tbody>
<tr>
<td>HSM</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
<tr>
<td>PLT</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
<tr>
<td>ALLEGO</td>
<td>27.07</td>
<td>25.07</td>
<td>23.07</td>
<td>0.70</td>
<td>0.50</td>
<td>0.30</td>
</tr>
</tbody>
</table>

Acknowledgments: Kalina Jasinska is supported by the Polish National Science Center under grant no. 2018/31/D/ST6/00305. The work of Marek Wydmuch was supported by the Polish Ministry of Science and Higher Education under grant no. 2017/26/N/ST6/01523. Computation resources on the PIONIER platform are provided by the Polish Supercomputing and Networking Center. The software is provided by:

Mark Wydmuch is supported by: