Online probabilistic label trees (OPLTs)

Kalina Jasinska-Kobus[*,1,3] Marek Wydmuch[*,1]
Devanathan Thiruvenkatachari[2] Krzysztof Dembczyński[1,2]

[1]Poznań University of Technology, Poland
[2]Yahoo! Research, New York, USA
[3]ML Research at Allegro.pl, Poland

allegro ML Research

The 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021.

[*]Equal contribution

Motivation

- In modern machine learning applications, the label space can be enormous, containing even millions of different labels or classes (**eXtreme Classification (XC)**).
- Selected examples of such problems:
  - content annotation for multimedia search (Dekel and Shamir, 2010),
  - recommendation: webpages-to-ads (Beygelzimer et al., 2009), ads-to-bid-words (Agrawal et al., 2013; Prabhu and Varma, 2014), users-to-items (Weston et al., 2013; Zhuo et al., 2020), queries-to-items (Medini et al., 2019), or items-to-queries (Chang et al., 2020).
- In these practical applications, **learning algorithms run in rapidly changing environments**.
- **The space of labels and features might grow over time** as new data points arrive.
- **Retraining a XC model** every time a new label is observed **is expensive**.
- **A need for XC algorithms that can efficiently adapt to the growing label and feature space**.

Online probabilistic label trees (OPLTs

- Label tree algorithms that use probabilistic classifiers are a popular solution to XC:
  - **probabilistic label trees (PLT)s** (Jasinska et al., 2016), **Parabel** (Prabhu et al., 2018), **extremeText** (Wydmuch et al., 2018), **Bonsai** (Khandagale et al., 2019), **AttentionXML** (You et al., 2019), **napkinXC** (Jasinska-Kobus et al., 2020).
- They reduce the original problem to a set of binary problems organized in a form of a rooted, leaf-labeled tree.
- Most of the above algorithms can update node classifiers incrementally with new examples.
- In all of them, the label tree is given before training of the node classifiers, thus **they are limited to initial set of labels**.
- We introduce **online probabilistic label trees (OPLTs)** that train a label tree classifier in a **fully online manner – the tree is constructed simultaneously with incremental training of node classifiers**, without any prior knowledge of labels or training data.

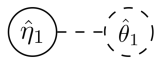
Online probabilistic label trees (OPLTs)

- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

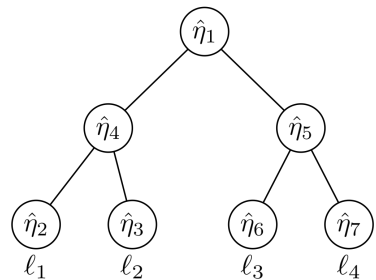
Updates:

$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.

OPLT



Incremental PLT



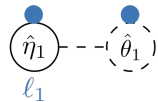
Online probabilistic label trees (OPLTs)

- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

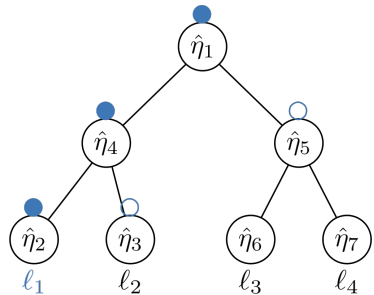
Updates:

$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.

OPLT



Incremental PLT

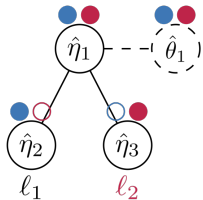


Online probabilistic label trees (OPLTs)

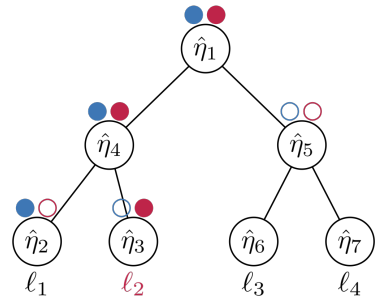
- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

Updates:
$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.

OPLT



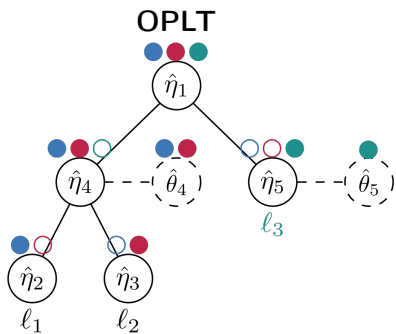
Incremental PLT

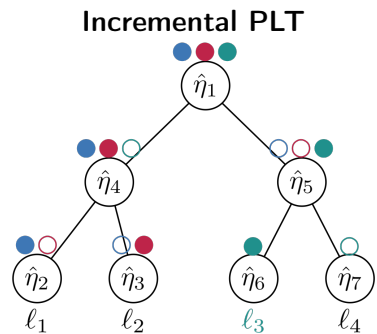


Online probabilistic label trees (OPLTs)

- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

Updates:
$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.

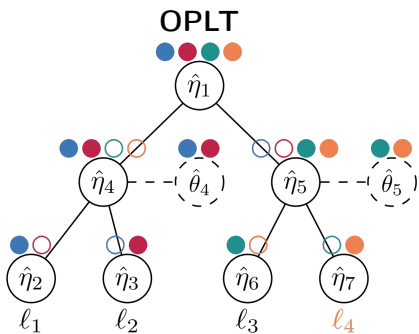


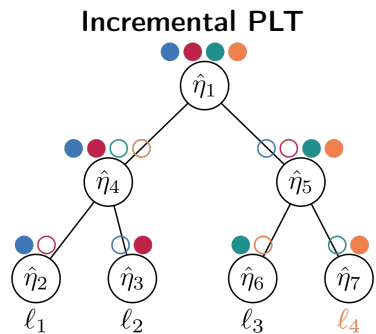


Online probabilistic label trees (OPLTs)

- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

Updates:
$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.

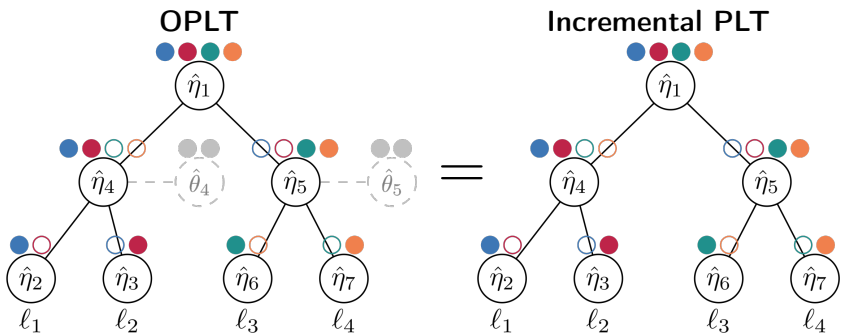




Online probabilistic label trees (OPLTs)

- Incremental PLT (IPLT) – trains node classifiers incrementally using a tree built by OPLT.
- Two properties of OPLT that we formalize and prove:
  - **efficiency**: the complexity of OPLT is in a constant factor of the complexity of IPLT.
  - **properness**: the final model trained by OPLT is equivalent to the model of IPLT.
- The properness is achieved thanks to **auxiliary node classifiers** that accumulate positive updates and are used to initialize classifiers in new nodes added to a tree.

Updates:
$(\boldsymbol{x}_1, \{l_1\})$,
$(\boldsymbol{x}_2, \{l_2\})$,
$(\boldsymbol{x}_3, \{l_3\})$,
$(\boldsymbol{x}_4, \{l_4\})$.



Thank you for your attention

References

Agrawal, R., Gupta, A., Prabhu, Y., and Varma, M. (2013). Multi-label learning with millions of labels: Recommending advertiser bid phrases for web pages. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, pages 13–24. International World Wide Web Conferences Steering Committee / ACM.

Beygelzimer, A., Langford, J., Lifshits, Y., Sorkin, G. B., and Strehl, A. L. (2009). Conditional probability tree estimation analysis and algorithms. In *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*, pages 51–58. AUAI Press.

Chang, W., Yu, H., Zhong, K., Yang, Y., and Dhillon, I. S. (2020). Taming pretrained transformers for extreme multi-label text classification. In Gupta, R., Liu, Y., Tang, J., and Prakash, B. A., editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3163–3171. ACM.

Dekel, O. and Shamir, O. (2010). Multiclass-multilabel classification with more classes than examples. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 137–144. JMLR.org.

Jasinska, K., Dembczynski, K., Busa-Fekete, R., Pfannschmidt, K., Klerx, T., and Hüllermeier, E. (2016). Extreme F-measure maximization using sparse probability estimates. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1435–1444. JMLR.org.

Jasinska-Kobus, K., Wydmuch, M., Dembczyński, K., Kuznetsov, M., and Busa-Fekete, R. (2020). Probabilistic label trees for extreme multi-label classification. *CoRR*, abs/2009.11218.

Khandagale, S., Xiao, H., and Babbar, R. (2019). Bonsai - diverse and shallow trees for extreme multi-label classification. *CoRR*, abs/1904.08249.

Medini, T. K. R., Huang, Q., Wang, Y., Mohan, V., and Shrivastava, A. (2019). Extreme classification in log memory using count-min sketch: A case study of amazon search with 50m products. In Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 13265–13275. Curran Associates, Inc.

Prabhu, Y., Kag, A., Harsola, S., Agrawal, R., and Varma, M. (2018). Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 993–1002. ACM.

Prabhu, Y. and Varma, M. (2014). Fastxml: a fast, accurate and stable tree-classifier for extreme multi-label learning. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 263–272. ACM.

Weston, J., Makadia, A., and Yee, H. (2013). Label partitioning for sublinear ranking. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 181–189. JMLR.org.

Wydmuch, M., Jasinska, K., Kuznetsov, M., Busa-Fekete, R., and Dembczynski, K. (2018). A no-regret generalization of hierarchical softmax to extreme multi-label classification. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 6355–6366. Curran Associates, Inc.

You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H., and Zhu, S. (2019). Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. In Wallach, H., Larochelle, H., Beygelzimer, A., d Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 5812–5822. Curran Associates, Inc.

Zhuo, J., Xu, Z., Dai, W., Zhu, H., Li, H., Xu, J., and Gai, K. (2020). Learning optimal tree models under beam search. In *Proceedings of the 37th International Conference on Machine Learning*, Vienna, Austria. PMLR.