CTU

CZECH TECHNICAL
UNIVERSITY
IN PRAGUE

Learning to Rank in Automatic Theorem Proving

Filip Bártek (filip.bartek@cvut.cz), Martin Suda

Czech Technical University in Prague

March 25, 2024

This research was supported by the Czech Science Foundation grant 24-12759S and COST Action CA20111 EuroProofNet.

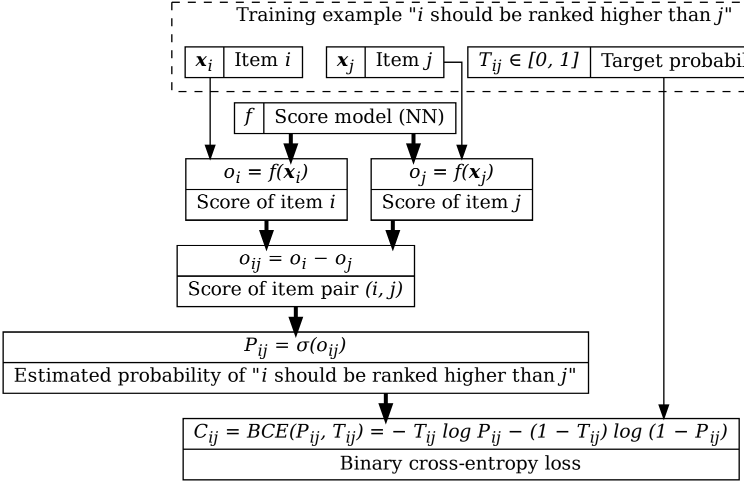
Motivation: Clause selection

- Goal: Train a *clause selection model*
  - Input: Set of clauses
  - Output (one of):
    1. Labeling of the input clauses (positive, negative)
    2. Best of the input clauses
    3. *Ranking of the input clauses*
- Training data (one of):
  1. Clauses with labels (positive, negative)
  2. Set of proof derivations. Each proof derivation is a set of clauses with labels (positive, negative).
  3. *Pairs of clauses C_+, C_- such that C_+ should be selected before C_-*

Learning to rank: Pairwise approach

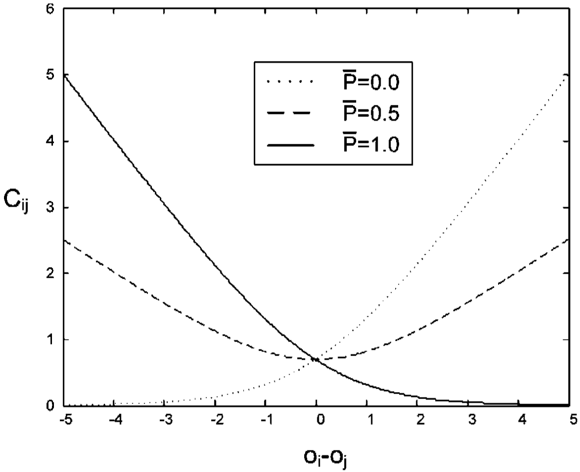
- Goal: Train a *ranking model*
  - Input: Set of items (samples, documents) D
  - Output: Ranking (permutation) over D
- Training example: Pair of items (i, j) such that i is to be ranked higher than j
- Main application domain: Recommender systems

RankNet



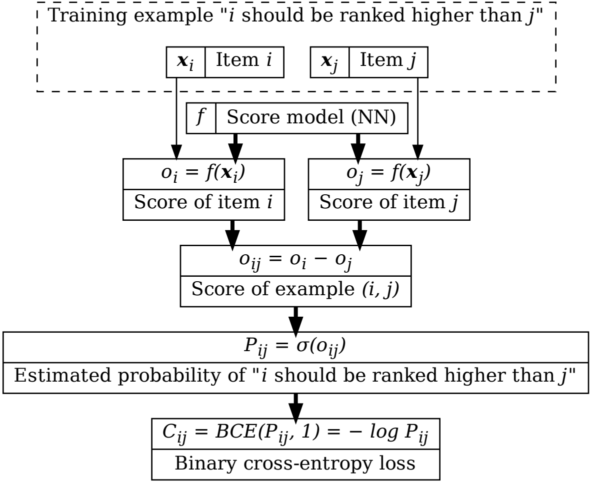
$$\sigma(o_{ij}) = \frac{e^{o_{ij}}}{1 + e^{o_{ij}}}$$

RankNet loss as a function of $o_i - o_j$



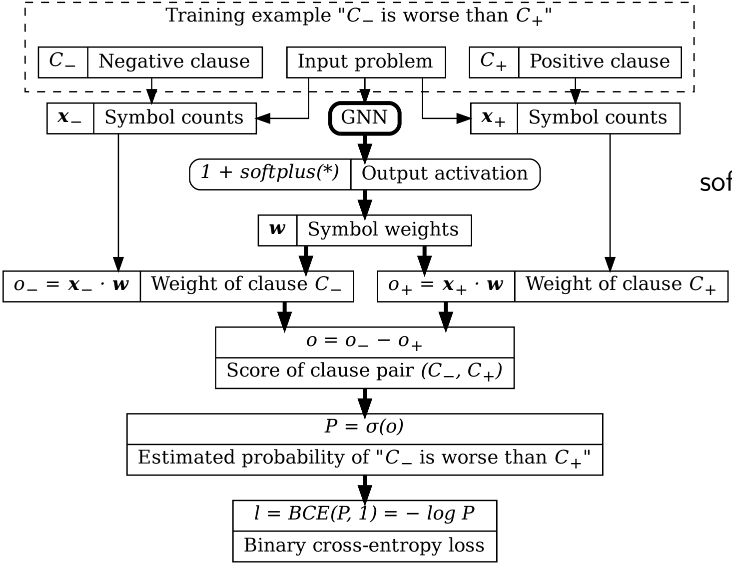
Credit: Burges et al. Learning to rank using gradient descent. ICML 2005.

RankNet with $T_{ij} = 1$



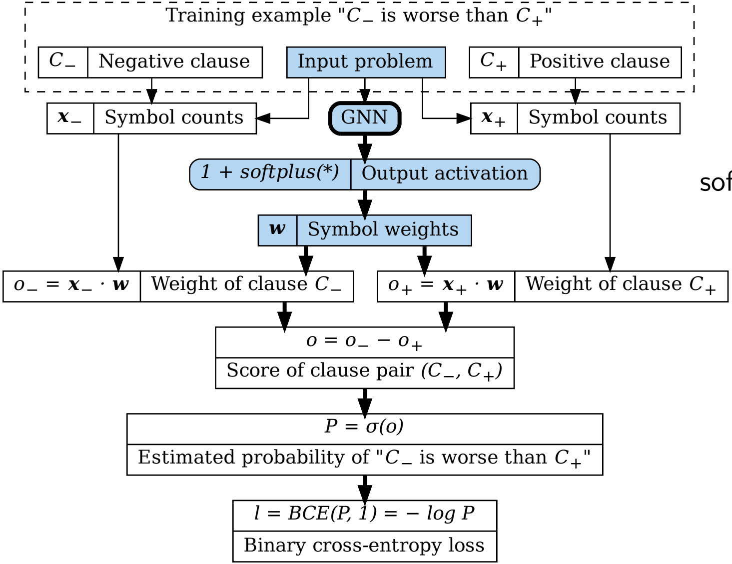
$$C_{ij} = \log(1 + e^{o_j - o_i})$$

Symbol weight recommender



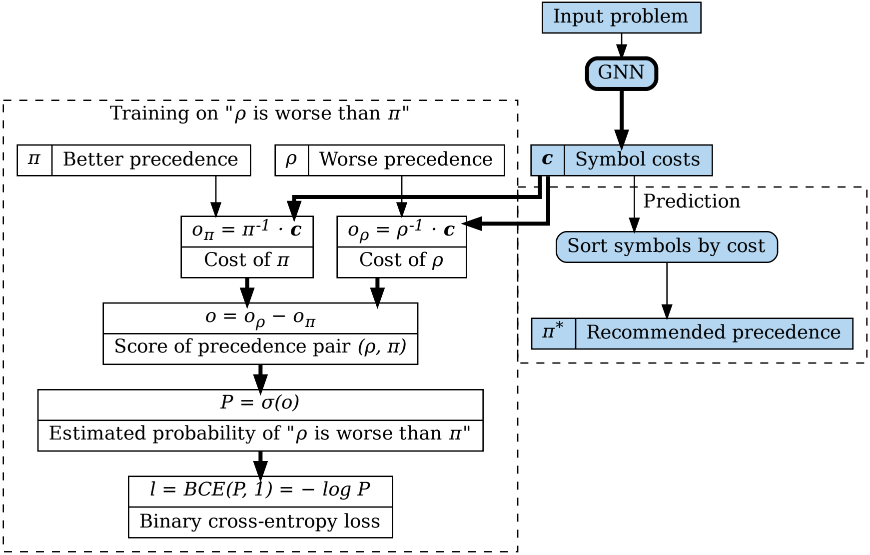
$$\mathrm{softplus}(x) = \log(1 + e^x)$$

Symbol weight recommender



$$\text{softplus}(x) = \log(1 + e^x)$$

Symbol precedence recommender



Conclusion

Tasks suitable for RankNet:

- Goal: Rank a set of items or get a top-ranked item
- Training data: Ranked pairs of items

Conclusion

Tasks suitable for RankNet:

- Goal: Rank a set of items or get a top-ranked item
- Training data: Ranked pairs of items

Future work with RankNet:

- Clause selection:
  - Train a full NN clause ranking model to be queried at runtime
  - Generalize symbol counting clause weight to a RNN on term structure
  - Optimize symbol weights on problems with a common signature, use logistic regression instead of gradient descent
- Simplification ordering on terms: Train KBO symbol weight jointly with precedence
- Stress top-ranked items more when training

Conclusion

Tasks suitable for RankNet:

- Goal: Rank a set of items or get a top-ranked item
- Training data: Ranked pairs of items

Future work with RankNet:

- Clause selection:
  - Train a full NN clause ranking model to be queried at runtime
  - Generalize symbol counting clause weight to a RNN on term structure
  - Optimize symbol weights on problems with a common signature, use logistic regression instead of gradient descent
- Simplification ordering on terms: Train KBO symbol weight jointly with precedence
- Stress top-ranked items more when training

Thank you for your attention!

Appendix

References

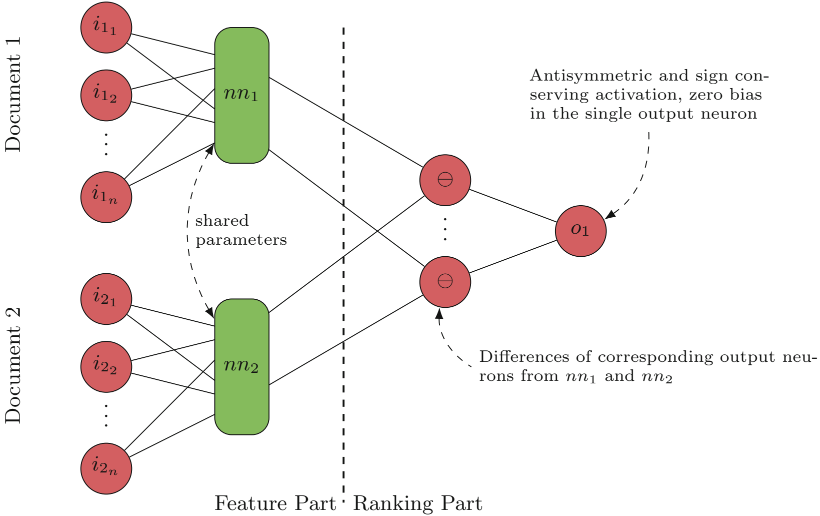
Filip Bártek and Martin Suda.
Neural precedence recommender.
In André Platzer and Geoff Sutcliffe, editors, *Automated Deduction - CADE 28 - 28th International Conference on Automated Deduction, Virtual Event, July 12-15, 2021, Proceedings*, volume 12699 of *Lecture Notes in Computer Science*, pages 525–542. Springer, 2021.

Filip Bártek and Martin Suda.
How much should this symbol weigh? A gnn-advised clause selection.
In Ruzica Piskac and Andrei Voronkov, editors, *LPAR 2023: Proceedings of 24th International Conference on Logic for Programming, Artificial Intelligence and Reasoning, Manizales, Colombia, 4-9th June 2023*, volume 94 of *EPiC Series in Computing*, pages 96–111. EasyChair, 2023.

Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N. Hullender.
Learning to rank using gradient descent.
In Luc De Raedt and Stefan Wrobel, editors, *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, volume 119 of *ACM International Conference Proceeding Series*, pages 89–96. ACM, 2005.

Marius Köppel, Alexander Segner, Martin Wagener, Lukas Pensel, Andreas Karwath, and Stefan Kramer.
Pairwise learning to rank by neural networks revisited: Reconstruction, theoretical analysis and practical performance.
In Ulf Brefeld, Élisa Fromont, Andreas Hotho, Arno J. Knobbe, Marloes H. Maathuis, and Céline Robardet, editors, *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part III*, volume 11908 of *Lecture Notes in Computer Science*, pages 237–252. Springer, 2019.

DirectRanker



Credit: Köppel et al. [4]

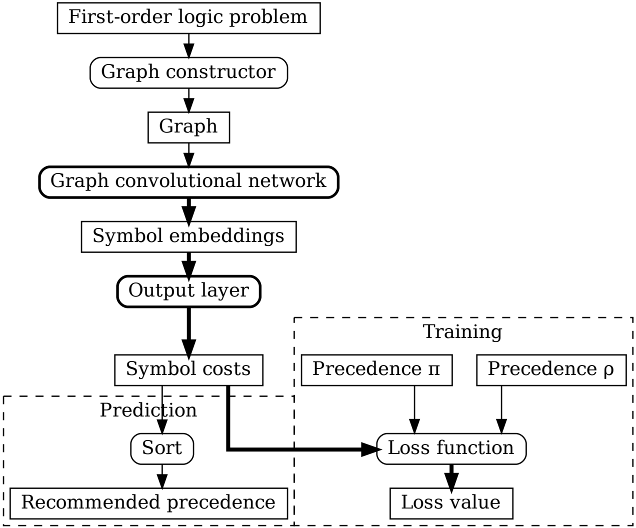
RankNet

Training example: Pair of items i, j and target probability $\bar{P}_{ij}$ of "$i \triangleright j$"
Loss function:

- $o_i = f(i)$ ... score of item i
- $o_{ij} = o_i - o_j$ ... score of pair of items i, j
- $P_{ij} = \sigma(o_{ij}) = \frac{e^{o_{ij}}}{1+e^{o_{ij}}}$ ... predicted probability of "$i \triangleright j$"
- $C_{ij} = -\bar{P}_{ij} \log P_{ij} - (1 - \bar{P}_{ij}) \log(1 - P_{ij}) = -\bar{P}_{ij} o_{ij} + \log(1 + e^{o_{ij}})$ ... binary cross-entropy loss

Properties: reflexive ($o_{ii} = 0$), antisymmetric ($o_{ij} = -o_{ji}$), transitive ($o_{ij} \geq 0 \wedge o_{jk} \geq 0 \implies o_{ik} \geq 0$)

Symbol precedence recommender: Overview



Symbol weight recommender: Overview