Abstract

This paper presents the computational challenge on topological deep learning that was hosted within the ICML 2023 Workshop on Topology and Geometry in Machine Learning. The competition asked participants to provide open-source implementations of topological neural networks from the literature by contributing to the python packages TopoNetX (data processing) and TopoModelX (deep learning). The challenge attracted twenty-eight qualifying submissions in its two-month duration. This paper describes the design of the challenge and summarizes its main findings. Code: https://github.com/pyt-team/TopoModelX. DOI: 10.5281/zenodo.7958513.

1. Introduction

Graph neural networks (GNNs) have proven to be a powerful deep learning architecture for processing relational data. More specifically, GNNs operate in graph domains comprised of pairwise relations between nodes. Topological neural networks (TNNs) extend GNNs by operating on domains featuring higher-order relations. Such domains, called topological domains, feature part-whole and/or set-type relations (Fig. 1) (Hajij et al., 2023), allowing a more expressive representation of the data. By operating on a topological domain, a TNN leverages the intricate relational structure at the heart of the data. Topological deep learning (Bodnar, 2022; Hajij et al., 2023) has shown great promise in many applications, ranging from molecular classification to social network prediction. However, the adoption of its architectures has been limited by the fragmented availability of open-source algorithms and lack of benchmarking between topological domains.

The challenge described in this white paper aims to fill that gap by implementing models in a unifying open-source software. In doing so, the challenge contributes to fostering reproducible research in topological deep learning. Participants were asked to contribute code for a published TNN, following TopoModelX’s API (Hajij et al., 2023) and computational primitives, and implement a training mechanism for the algorithm’s intended task.

This white paper is organized as follows. Section 2 describes the setup of the challenge, including its guidelines and evaluation criteria. Section 3 lists all qualifying submissions to the challenge and its winners.

2. Setup of the challenge

The challenge was held in conjunction with the workshop Topology and Geometry in Machine Learning of the International Conference on Machine Learning (ICML) 2023. Participants were asked to contribute code for a previously existing TNN and train it on a toy dataset of their choice.
Guidelines Each submission took the form of an implementation of a pre-existing TNN listed in a survey of the field (Papillon et al., 2023). These models fall into four categories, defined by their topological domain. All submitted code was required to comply with TopoModelX’s GitHub Action workflow (Hajij et al., 2023), successfully passing all tests, linting, and formatting.

Each submission consisted of a pull request to TopoModelX containing three new files:

1. A Python script implementing a layer of the model in a single class using TopoModelX computational primitives. One layer is equivalent to the message passing depicted in the tensor diagram representation for the model given in the survey (Papillon et al., 2023).

2. A Jupyter notebook that builds a neural network out of the single layer, loads and pre-processes the chosen dataset, and performs a train-test loop on the dataset. Defining training and testing in a Jupyter notebook offers authors a natural way to communicate results that are reproducible, as anyone with access to the notebook may run it to attain analogous results.

3. A Python script which contains the unit tests for all methods stored in the class defining the model layer.

Teams were registered to the challenge upon submission of their pull request and there was no restriction on the number of team members, nor on the amount of submissions per team.

The principal developers of TopoModelX were not allowed to participate. Consistent with the aims of an open environment for sharing participation in this activity is completely voluntary and no support or endorsement of any of the participating parties by any of the other participating parties is provided. All submissions are the views of the individual participants only and should be taken, as is with all faults and without any guarantee, promise or endorsement of any kind.

Evaluation criteria The evaluation criteria were:

1. Does the submission implement the chosen model correctly, specifically in terms of its message passing scheme? (The training schemes do not need to match that of the original model).

2. How readable and clean is the code? How well does the submission respect TopoModelX’s APIs?

3. Is the submission well-written? Do the docstrings clearly explain the methods? Are the unit tests robust?

Note that these criteria were not designed to reward model performance, nor complexity of training. Rather, these criteria aimed to reward clean code and accurate model architectures that will foster reproducible research in topological deep learning.

Evaluation Method The Condorcet method (Young, 1988) was used to rank the submissions and decide on the winners. Each team whose submission respected the guidelines was given one vote in the decision process. Nine additional reviewers selected from PyT-team maintainers and collaborators were also each given a vote. Upon voting, participating teams and reviewers were each asked to select the best and second best model implementation in each topological domain, thus making eight choices in total. Participants were not allowed to vote for their own submissions.

Software engineering practices Challenge participants were encouraged to use software engineering best practices. All code had to be compatible with Python 3.10 and a reasonable effort had to be made for the code to adhere to PEP8 Python style guidelines. The chosen dataset
had to be loaded from TopoNetX (Hajij et al., 2023) or PyTorch-Geometric (Fey & Lenssen, 2019). Participants could raise GitHub issues and/or request help at any time by contacting the organizers.

3. Submissions and Winners

In total, the challenge received 32 submissions, 28 of which adhered to the above outlined qualification requirements. Out of the qualifying submissions, 23 unique models were implemented. All four topological domains are represented in this set of models: 12 hypergraph implementations, 11 simplicial model implementations, 3 cellular implementations, and 2 combinatorial implementations.

Table 4 lists all qualifying submissions. (Papillon et al., 2023) contains additional information on the architectures and message-passing frameworks for each of these models.

Table 4 also indicates the winning contributions, consisting of a first and second prize for each topological domain, as well as honorable mentions. The winners were announced publicly at the ICML Workshop on Topology, Algebra and Geometry in Machine Learning and on social medias. Regardless of this final ranking, we would like to stress that all the submissions were of very high quality. We warmly congratulate all participants.

4. Conclusion

This white paper presented the motivation and outcomes of the organization of the Topological Deep Learning Challenge hosted through the ICML 2023 workshop on Topology, Algebra and Geometry in Machine Learning. Challenge submissions implemented a wide variety of topological neural networks into the open-source package TopoModelX. We hope that this community effort will foster reproducible research and further methodological benchmarks in the growing field of topological deep learning.

Acknowledgments

The authors would like the thank the organizers of the ICML 2023 Topology, Algebra and Geometry in Machine Learning Workshop for their valuable support in the organization of the challenge.
<table>
<thead>
<tr>
<th>Domain</th>
<th>Model</th>
<th>Task Level</th>
<th>Computational challenge submission authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>HG</td>
<td>HyperSage (Arya et al., 2020)</td>
<td>✓</td>
<td>German Magai, Pavel Snopov</td>
</tr>
<tr>
<td></td>
<td>AllSetTransformer (Chien et al., 2022)</td>
<td>✓</td>
<td>Luca Scnofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernard (first place)</td>
</tr>
<tr>
<td></td>
<td>AllSetTransformer (Chien et al., 2022)</td>
<td>✓</td>
<td>Luca Scnofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernard (first place)</td>
</tr>
<tr>
<td></td>
<td>HyperGat (Ding et al., 2020)</td>
<td>✓</td>
<td>German Magai, Pavel Snopov</td>
</tr>
<tr>
<td></td>
<td>HNHN (Dong et al., 2020)</td>
<td>✓ ✓</td>
<td>1. Alessandro Salatiello (hon. mention) 2. Sadrodin Barikbin</td>
</tr>
<tr>
<td></td>
<td>HMPNN* (Heydari &amp; Livi, 2022)</td>
<td>✓</td>
<td>Sadrodin Barikbin (second place)</td>
</tr>
<tr>
<td></td>
<td>UniGCN (Huang &amp; Yang, 2021)</td>
<td>✓</td>
<td>Alexander Nikitin (hon. mention)</td>
</tr>
<tr>
<td></td>
<td>UniSAGE (Huang &amp; Yang, 2021)</td>
<td>✓</td>
<td>Alexander Nikitin</td>
</tr>
<tr>
<td></td>
<td>UniGCNII (Huang &amp; Yang, 2021)</td>
<td>✓</td>
<td>Paul Häusner, Jens Sjölund</td>
</tr>
<tr>
<td></td>
<td>UniGIN (Huang &amp; Yang, 2021)</td>
<td>✓</td>
<td>Kalyan Nadimpalli</td>
</tr>
<tr>
<td></td>
<td>DHGCN* (Wei et al., 2021)</td>
<td>✓</td>
<td>Tatiana Malygina</td>
</tr>
<tr>
<td>SC</td>
<td>SCCONV (Bunch et al., 2020)</td>
<td>✓</td>
<td>Abdelwahed Khamis, Ali Zia, Mohammed Hassanin</td>
</tr>
<tr>
<td></td>
<td>SNN (Ebli et al., 2020)</td>
<td>✓</td>
<td>Jens Agerberg, Georg Bökman, Pavlo Melnyk</td>
</tr>
<tr>
<td></td>
<td>SAN (Giusti et al., 2022a)</td>
<td>✓</td>
<td>Luca Scnofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernard (first place)</td>
</tr>
<tr>
<td></td>
<td>SCA (Hajij et al., 2022a)</td>
<td>✓</td>
<td>Aiden Brent (hon. mention)</td>
</tr>
<tr>
<td></td>
<td>Dist2Cycle (Keros et al., 2022)</td>
<td>✓</td>
<td>Ali Zia</td>
</tr>
<tr>
<td></td>
<td>SCoNe (Roddenberry et al., 2021)</td>
<td>✓</td>
<td>1. Odin Hoff Gardaa (second place) 2. Aiden Brent</td>
</tr>
<tr>
<td></td>
<td>SCN (Yang et al., 2022a)</td>
<td>✓</td>
<td>Maosheng Yang, Lucia Testa</td>
</tr>
<tr>
<td></td>
<td>SCN (Yang et al., 2022b)</td>
<td>✓</td>
<td>Yixiao Yue</td>
</tr>
<tr>
<td>CC</td>
<td>CWN (Bodnar et al., 2021)</td>
<td>✓ ✓</td>
<td>Dmitrii Gavrilev, Gleb Bazhenov, Suraj Singh (second place)</td>
</tr>
<tr>
<td></td>
<td>CAN (Giusti et al., 2022b)</td>
<td>✓</td>
<td>1. Luca Scnofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernard (first place) 2. Abraham Rabinowitz</td>
</tr>
<tr>
<td>CCC</td>
<td>HOAN (Hajij et al., 2022b)</td>
<td>✓ ✓</td>
<td>1. Rubén Ballester, Manuel Lecha, Sergio Escalera (first place) 2. Aiden Brent (second place)</td>
</tr>
</tbody>
</table>

Table 1. Model implementations submitted to the Topological Deep Learning Challenge. We organize original models according to domain: hypergraph (HG), simplicial (SC), cellular (CC), and combinatorial (CCC). Task level indicates the rank on which a prediction is made.
References


Notes