ICML 2023 Topological Deep Learning Challenge: Design and Results

Mathilde Papillon [* 1] **Mustafa Hajij** [* 1 2] **Helen Jenne** [2 3] **Johan Mathe** [2 3] **Audun Myers** [2 3]
Theodore Papamarkou [2 3] **Tolga Birdal** [3] **Tamal Dey** [3] **Tim Doster** [3] **Tegan Emerson** [3]
Gurusankar Gopalakrishnan [3] **Devendra Govil** [3] **Aldo Guzmán-Sáenz** [3] **Henry Kvinge** [3] **Neal Livesay** [3]
Soham Mukherjee [3] **Shreyas N. Samaga** [3] **Karthikeyan Natesan Ramamurthy** [3] **Maneel Reddy Karri** [3]
Paul Rosen [3] **Sophia Sanborn** [3] **Robin Walters** [3] **Jens Agerberg** [4] **Sadrodin Barikbin** [4] **Claudio Battiloro** [4]
Gleb Bazhenov [4] **Guillermo Bernardez** [4] **Aiden Brent** [4] **Sergio Escalera** [4] **Simone Fiorellino** [4] **Dmitrii Gavrilev** [4]
Mohammed Hassanin [4] **Paul Häusner** [4] **Odin Hoff Gardaa** [4] **Abdelwahed Khamis** [4] **Manuel Lecha** [4]
German Magai [4] **Tatiana Malygina** [4] **Rubén Ballester** [4] **Kalyan Nadimpalli** [4] **Alexander Nikitin** [4]
Abraham Rabinowitz [4] **Alessandro Salatiello** [4] **Simone Scardapane** [4] **Luca Scofano** [4] **Suraj Singh** [4]
Jens Sjölund [4] **Pavel Snopov** [4] **Indro Spinelli** [4] **Lev Telyatnikov** [4] **Lucia Testa** [4] **Maosheng Yang** [4] **Yixiao Yue** [4]
Olga Zaghen [4] **Ali Zia** [4] **Nina Miolane** [1]

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 [1] was held in conjunction with the workshop Topology and Geometry in Machine Learning of the International Conference on Machine Learning (ICML) 2023 [2]. Participants were asked to contribute code for a previously existing TNN and train it on a toy dataset of their choice.

[*]Equal contribution [1]Challenge Organizer [2]Challenge Reviewer [3]Challenge Contributor [4]Challenge Participant. Correspondence to: Mathilde Papillon <papillon@ucsb.edu>.

Proceedings of the 2^{nd} Topology and Geometry in Machine Learning Workshop at the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA.

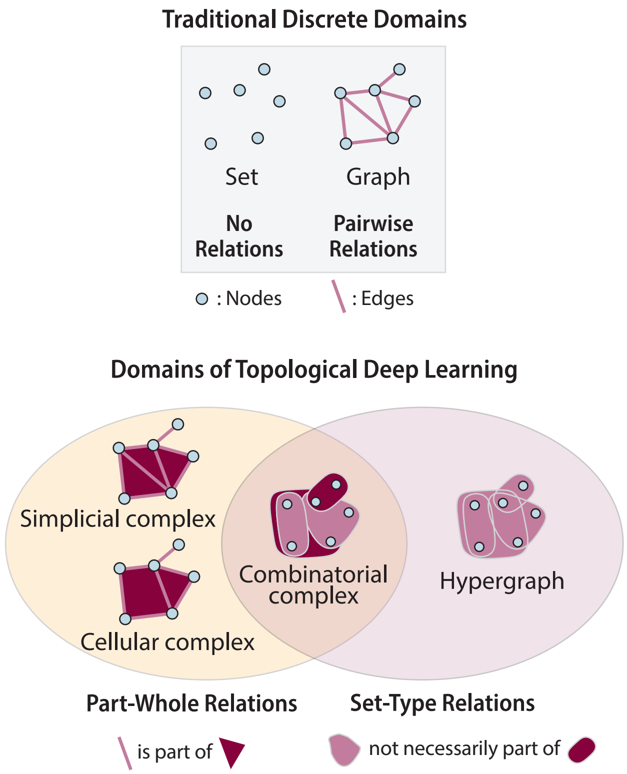

Figure 1. **Domains:** Nodes in light blue, (hyper)edges in pink, and faces in dark red. Adapted from (Hajij et al., 2023).

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.

Domain	Model	Task Level: *Node*	*Edge*	*Complex*	Computational challenge submission authors
HG	HyperSage (Arya et al., 2020)	✓			German Magai, Pavel Snopov
	AllSetTransformer (Chien et al., 2022)	✓			Luca Scofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernardez (first place)
	AllSetTransformer (Chien et al., 2022)	✓			Luca Scofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernardez
	HyperGat (Ding et al., 2020)	✓			German Magai, Pavel Snopov
	HNHN (Dong et al., 2020)	✓	✓		1. Alessandro Salatiello (hon. mention) 2. Sadrodin Barikbin
	HMPNN* (Heydari & Livi, 2022)	✓			Sadrodin Barikbin (second place)
	UniGCN (Huang & Yang, 2021)	✓			Alexander Nikitin (hon. mention)
	UniSAGE (Huang & Yang, 2021)	✓			Alexander Nikitin
	UniGCNII (Huang & Yang, 2021)	✓			Paul Häusner, Jens Sjölund
	UniGIN (Huang & Yang, 2021)	✓			Kalyan Nadimpalli
	DHGCN* (Wei et al., 2021)			✓	Tatiana Malygina
SC	SCCONV (Bunch et al., 2020)			✓	Abdelwahed Khamis, Ali Zia, Mohammed Hassanin
	SNN (Ebli et al., 2020)		✓		Jens Agerberg, Georg Bökman, Pavlo Melnyk
	SAN (Giusti et al., 2022a)		✓		Luca Scofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernardez (first place)
	SCA (Hajij et al., 2022a)			✓	Aiden Brent (hon. mention)
	Dist2Cycle (Keros et al., 2022)		✓		Ali Zia
	SCoNe (Roddenberry et al., 2021)		✓		1. Odin Hoff Gardaa (second place) 2. Aiden Brent
	SCNN (Yang et al., 2022a)		✓		Maosheng Yang, Lucia Testa
	SCCNN (Yang & Isufi, 2023)		✓		1. Maosheng Yang, Lucia Testa 2. Jens Agerberg, Georg Bökman, Pavlo Melnyk (hon. mention)
	SCN (Yang et al., 2022b)		✓		Yixiao Yue
CC	CWN (Bodnar et al., 2021)		✓	✓	Dmitrii Gavrilev, Gleb Bazhenov, Suraj Singh (second place)
	CAN (Giusti et al., 2022b)			✓	1. Luca Scofano, Indro Spinelli, Simone Scardapane, Simone Fiorellino, Olga Zaghen, Lev Telyatnikov, Claudio Battiloro, Guillermo Bernardez (first place) 2. Abraham Rabinowitz
CCC	HOAN (Hajij et al., 2022b)		✓	✓	1. Rubén Ballester, Manuel Lecha, Sergio Escalera (first place) 2. Aiden Brent (second place)

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

Arya, D., Gupta, D. K., Rudinac, S., and Worring, M. Hypersage: Generalizing inductive representation learning on hypergraphs. *arXiv preprint arXiv:2010.04558*, 2020.

Bodnar, C. *Topological Deep Learning: Graphs, Complexes, Sheaves*. PhD thesis, Apollo - University of Cambridge Repository, 2022. URL https://www.repository.cam.ac.uk/handle/1810/350982.

Bodnar, C., Frasca, F., Otter, N., Wang, Y., Lio, P., Montufar, G. F., and Bronstein, M. Weisfeiler and Lehman Go Cellular: CW Networks. *Advances in Neural Information Processing Systems*, 34:2625–2640, 2021.

Bunch, E., You, Q., Fung, G., and Singh, V. Simplicial 2-complex convolutional neural networks. In *TDA & Beyond*, 2020.

Chien, E., Pan, C., Peng, J., and Milenkovic, O. You are allset: A multiset function framework for hypergraph neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=hpBTIv2uy_E.

Ding, K., Wang, J., Li, J., Li, D., and Liu, H. Be more with less: Hypergraph attention networks for inductive text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4927–4936, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.399. URL https://aclanthology.org/2020.emnlp-main.399.

Dong, Y., Sawin, W., and Bengio, Y. Hnhn: Hypergraph networks with hyperedge neurons. *ICML Graph Representation Learning and Beyond Workshop*, 2020. URL https://arxiv.org/abs/2006.12278.

Ebli, S., Defferrard, M., and Spreemann, G. Simplicial neural networks. In *TDA & Beyond*, 2020.

Fey, M. and Lenssen, J. E. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

Giusti, L., Battiloro, C., Di Lorenzo, P., Sardellitti, S., and Barbarossa, S. Simplicial attention networks. *arXiv preprint arXiv:2203.07485*, 2022a.

Giusti, L., Battiloro, C., Testa, L., Di Lorenzo, P., Sardellitti, S., and Barbarossa, S. Cell attention networks. *arXiv preprint arXiv:2209.08179*, 2022b.

Hajij, M., Zamzmi, G., Papamarkou, T., Maroulas, V., and Cai, X. Simplicial complex representation learning. In *Machine Learning on Graphs (MLoG) Workshop at 15th ACM International WSDM (2022) Conference, WSDM2022-MLoG ; Conference date: 21-02-2022 Through 25-02-2022*, January 2022a.

Hajij, M., Zamzmi, G., Papamarkou, T., Miolane, N., Guzmán-Sáenz, A., and Ramamurthy, K. N. Higher-order attention networks. *arXiv preprint arXiv:2206.00606*, 2022b.

Hajij, M., Zamzmi, G., Papamarkou, T., Miolane, N., Guzmán-Sáenz, A., Ramamurthy, K. N., Birdal, T., Dey, T. K., Mukherjee, S., Samaga, S. N., Livesay, N., Walters, R., Rosen, P., and Schaub, M. T. Topological deep learning: Going beyond graph data, 2023.

Heydari, S. and Livi, L. Message passing neural networks for hypergraphs. In Pimenidis, E., Angelov, P. P., Jayne, C., Papaleonidas, A., and Aydin, M. (eds.), *Proceedings of 31st International Conference on Artificial Neural Networks, Part II*, volume 13530 of *Lecture Notes in Computer Science*, pp. 583–592. Springer, 2022. doi: 10.1007/978-3-031-15931-2_48.

Huang, J. and Yang, J. Unignn: a unified framework for graph and hypergraph neural networks. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 2021.

Keros, A. D., Nanda, V., and Subr, K. Dist2cycle: A simplicial neural network for homology localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7133–7142, 2022.

Papillon, M., Sanborn, S., Hajij, M., and Miolane, N. Architectures of topological deep learning: A survey on topological neural networks. *arXiv preprint arXiv:2304.10031*, 2023.

Roddenberry, T. M., Glaze, N., and Segarra, S. Principled simplicial neural networks for trajectory prediction. In *International Conference on Machine Learning*, pp. 9020–9029. PMLR, 2021.

Wei, J., Wang, Y., Guo, M., Lv, P., Yang, X., and Xu, M. Dynamic hypergraph convolutional networks for skeleton-based action recognition. *arXiv preprint arXiv:2112.10570*, 2021.

Yang, M. and Isufi, E. Convolutional learning on simplicial complexes. *arXiv preprint arXiv:2301.11163*, 2023.

Yang, M., Isufi, E., Schaub, M. T., and Leus, G. Simplicial convolutional filters. *IEEE Transactions on Signal Processing*, 70:4633–4648, 2022a.

Yang, R., Sala, F., and Bogdan, P. Efficient representation learning for higher-order data with simplicial complexes. In *Learning on Graphs Conference*, pp. 13–1. PMLR, 2022b.

Young, H. P. Condorcet's theory of voting. *American Political science review*, 82(4):1231–1244, 1988.

Notes

1. Challenge website: https://pyt-team.github.io/topomodelx/challenge/index.html

2. Topology and Geometry in Machine Learning Workshop website: https://www.tagds.com/events/conference-workshops/tag-ml23