1 Introduction

The original paper proposes a novel approach to learning on geometric graphs and point clouds that is equivariant to the euclidean group E(n) (rotations, translations, and reflections); the method is called Equivariant Message Passing Simplicial Networks (EMPSNs). This work uses the most common type of Graph Neural Networks (GNNs): Message Passing Neural Networks (MPNNs) for graph learning tasks where equivariance is important (e.g. molecular property prediction).

Message passing neural networks (MPNNs) have a number of drawbacks: Firstly the MPNNs are limited in learning higher dimensional graph structures such as cliques. Secondly, it is well known deep message passing neural networks can cause over-smoothing, after many local neighborhood aggregations, features of all nodes converge to the same value leading to indistinguishable node representations. EMPSN is designed to combat both of these well-known drawbacks of message passing neural networks. The idea of EMPSNs is therefore to consider a more elaborate topological space by incorporating a graph's higher dimensional simplices as learnable features.

Imagine a molecular graph, the molecule as a whole has a number of quantifiable properties and each atom (node) has geometric information. In EMPSN not only do the nodes pass messages to one another but simplices created from the node in the graph also pass messages to one another. These simplices have E(n) equivariant geometric features information derived from the geometric information of the nodes contained within this simplex. In this way, higher dimensional graph structures are explicitly represented in the message passing framework. EMPSNs are applicable to geometric graph data and can be used to predict graph level features.

The methodology starts by performing a graph lift, To limit the amount of simplices constructed a Vietoris-Rips lift is applied. The simplices this yields are added as nodes to the graph, at this point the simplices are purely topological objects, a geometric realization is achieved by calculating a number of E(n) equivariant features based on the geometric information of the nodes contained in the simplex. Examples are, relative distance of nodes, angles between nodes and volume of the simplex. The higher order the simplex the more features are able to be calculated.

After this step the molecule is enriched with geometrically embedded higher order structures in the form of line segments and triangles. In the training of the network, messages are passed between nodes and between simplices according to a hardcoded schema. 0 dimensional simplices communicate with 0 dimensional and 1 dimensional simplices. 1 dimensional simplices communicate with 1 dimensional simplices and 2 dimensional simplices. 2 dimensional simplices only communicate with 1 dimensional simplices.

This methodology is novel since it is the only work that combines simplicial complexes with E(n) equivariant message passing in GNNs. There two branches of Graph neural network research that is closely related to the paper in question. Firstly, Message passing simplical networks (MPSNs). Besides, it also enriches the topology of the input graphs with simplices the difference with this work is that the feature information of the added simplices is not necessarily equivariant under any transformation group. Another related branch of work are Equivariant Graph neural networks, one paper in particular is mentioned in the EMPSN paper. In this paper, the weighted sum of all of relative differences is used in multiple steps of the message passing to make the node updates E(n) equivariant. EMPSN essentially borrowed ideas from both branches to construct a new methodology. We acknowledge that the choice for topological complex is an important one since it is an assumption of shape of the higher order structure in the data.

2 Background

Message passing neural networks have observed an increased popularity since their proposal by Gilmer et al. in 2017. In this blogpost, we will see how message passing techniques are used in relation with simplicial networks. We introduce the relevant definitions of message passing, simplicial complexes, equivariant message passing networks and message passing simplicial networks.

2.1 Message passing

Let $G = (V, E)$ be a graph consisting of nodes V and edges E. Then let each node $v_i \in V$ and edge $e_{ij} \in E$ have an associated node feature $\mathbf{f}_i \in \mathbb{R}^{c_n}$ and edge feature $\mathbf{a}_{ij} \in \mathbb{R}^{c_e}$, with dimensionality $c_n, c_e \in \mathbb{N}_{>0}$. The intuition behind message passing is that nodes have hidden states and at each iteration, we update them using the following formulas:

$\mathbf{m}_{ij} = \phi_m\left(\mathbf{f}_i, \mathbf{f}_j, \mathbf{a}_{ij}\right)$	Find messages from v_j to v_i
$\mathbf{m}_i = \underset{j \in \mathcal{N}(i)}{\mathrm{Agg}}\ \mathbf{m}_{ij}$	Aggregate messages to v_i, with Agg being any permutation invariant function over the neighbours
$\mathbf{f}_i' = \phi_f\left(\mathbf{f}_i, \mathbf{m}_i\right)$	Update hidden state $\mathbf{f}_i$

given that $\mathcal{N}(i)$ is the set of neighbours of node v_i and that ϕ_m and ϕ_f are parametrized by multilayer perceptrons. At the end, a permutation invariant aggregation is applied to all final hidden states of the nodes, in order to get a hidden state that represents the entire graph (commonly a scalar).

2.2 Simplicial complexes

A simplex in Geometry is defined as an extention of the concept of triangles to multiple dimensions. From this notion, a n-simplex consists of $n+1$ fully connected points. We consider three specific simplices:

- 0-simplex represents a point or a node
- 1-simplex represents a line or edge connecting two points
- 2-simplex is a triangle
- 3-simplex is a tetrahedron

To assign properties to higher-dimensional simplices within a graph, we utilize abstract simplicial complexes.

Extending this concept to a more general one, we define an **abstract simplicial complex** (ASC), which is a set of simplices, such that if an higher-dimensional simplex is in the ASC also all lower-dimensional simplices belonging to it are in the ASC. More formally, we define an ASC $\mathcal{K}$ is a collection of non-empty finite subsets of some set $\mathcal{V}$ such that for every set $\mathcal{T} \in \mathcal{K}$ and any non-empty subset $\mathcal{R} \subseteq \mathcal{T}$, it is the case that $\mathcal{R} \in \mathcal{K}$. Thanks to this definition, we can associate a set of points of some graph $\mathcal{G}$ to a higher-order structure, using a so-called **lifting transformation**. When applied to a graph $\mathcal{G}$, results in the ASC $\mathcal{K}$ with the property that if the vertices $\{v_0, ..., v_k\}$ form a clique in $\mathcal{G}$, then the simplex $\{v_0, ..., v_k\} \in \mathcal{K}$. In other words, if the vertices $\{v_0, ..., v_k\}$ in $\mathcal{G}$ have the property that every two distinct vertices are adjacent, then the simplex $\{v_0, ..., v_k\}$ is in the ASC $\mathcal{K}$ (Bodnar et al. 2021).

2.3 Equivariant message passing networks

Firstly, is important to define what equivariance is. Suppose G is a group and X and Y are sets on which G acts. A function $f : X \to Y$ is called equivariant with respect to G if it commutes with the group action. In other words, applying a transformation $g \in G$ followed by the function f yields the same result as first applying f and then the transformation. Formally,

$$\begin{aligned} f(g \cdot x) &= g \cdot f(x) && \text{equivariance} \\ f(g \cdot x) &= f(x) && \text{invariance} \end{aligned}$$

A frequently utilized model for geometric graphs is the $E(n)$ Equivariant Graph Neural Network (EGNN). This model enhances the message-passing process by incorporating positional data while maintaining equivariance to $E(n)$ (Satorras et al., 2021). This is crucial because in certain scenarios, the nodes within a graph are situated in a Euclidean space, creating what is known as a geometric graph. This spatial information can be integrated into the message-passing framework to incorporate physical attributes to leverage geometric data. The message function is adapted to depend on $E(n)$ invariant information, such as the distance between two nodes. Consequently, the initial step in the message-passing process is modified as follows

$$\mathbf{m}_{ij} = \phi_m \left(\mathbf{f}_i, \mathbf{f}_j, \mathrm{lnv}\left(\mathbf{x}_i, \mathbf{x}_j\right), \mathbf{a}_{ij}\right)$$

having that for all $g \in E(n)$

$$\mathrm{lnv}\left(g \cdot \mathbf{x}_i, g \cdot \mathbf{x}_j\right) = \mathrm{lnv}\left(\mathbf{x}_i, \mathbf{x}_j\right)$$

Then for every layer the position of the nodes is updated as follows

$$\mathbf{x}_i' = \mathbf{x}_i + C \sum_{j \neq i} \left(\mathbf{x}_i - \mathbf{x}_j\right) \phi_x \left(\mathbf{m}_{ij}\right)$$

where ϕ_x is a MLP.

2.4 Message passing simplicial networks

We denote $\sigma \prec \tau$ when a simple σ is on the boundary of a simplex τ if an only if $\sigma \subset \tau$ and there is no δ such that $\sigma \subset \delta \subset \tau$. In other words, σ is directly on the boundary (immediately adjacent) of τ with any intermediate simplices between them. Message Passing Simplicial Networks (MPSNs) offer a message-passing framework that considers more complex forms of adjacency between objects in an abstract simplicial complex (ASC). Specifically, the following types of adjacency are identified:

- Boundary adjacencies $\mathcal{B}(\sigma) = \{\tau \mid \tau \prec \sigma\}$
- Co-boundary adjacencies $\mathcal{C}(\sigma) = \{\tau \mid \sigma \prec \tau\}$
- Lower-adjacencies $\mathcal{N}_{\downarrow}(\sigma) = \{\tau \mid \exists \delta, \delta \prec \tau \wedge \delta \prec \sigma\}$
- Upper-adjacencies $\mathcal{N}_{\uparrow}(\sigma) = \{\tau \mid \exists \delta, \tau \prec \delta \wedge \sigma \prec \delta\}$

In the case of simplicial complex, if it is a graph, then the upper adjacencies of a node $v \in \mathcal{V}$ is the set of nodes u that form an edge with v, defined as $\mathcal{N}_{\uparrow}(v) = \{u \in \mathcal{V} \mid \{v, u\} \in \mathcal{E}\}$. Then, as we saw in 2.1, the messages have to be aggregated to a simplex σ:

$$\mathbf{m}_{\mathcal{B}}(\sigma) = \underset{\tau \in \mathcal{B}(\sigma)}{\operatorname{Agg}} \left(\phi_{\mathcal{B}}\left(\mathbf{f}_\sigma, \mathbf{f}_\tau\right)\right)$$

having, again, that $\phi_{\mathcal{B}}$ is some MLP. Alternatively, we can incorporate four message types in the updates as follows:

$$\mathbf{f}'_\sigma = \phi_f\left(\mathbf{f}_\sigma, \mathbf{m}_{\mathcal{B}}(\sigma), \mathbf{m}_{\mathcal{C}}(\sigma), \mathbf{m}_{\mathcal{N}_\downarrow}(\sigma), \mathbf{m}_{\mathcal{N}_\uparrow}(\sigma)\right).$$

Lastly, for a k dimensional simplicial complex $\mathcal{K}$

$$\mathbf{h}_{\mathcal{K}} := \bigoplus_{i=0}^{k} \operatorname{Agg}_{\sigma \in \mathcal{K}, |\sigma| = i+1} \mathbf{h}_\sigma$$

where $\mathbf{h}_{\mathcal{K}}$ is the hidden state representing the entire complex, and $\bigoplus$ is the concatenation.

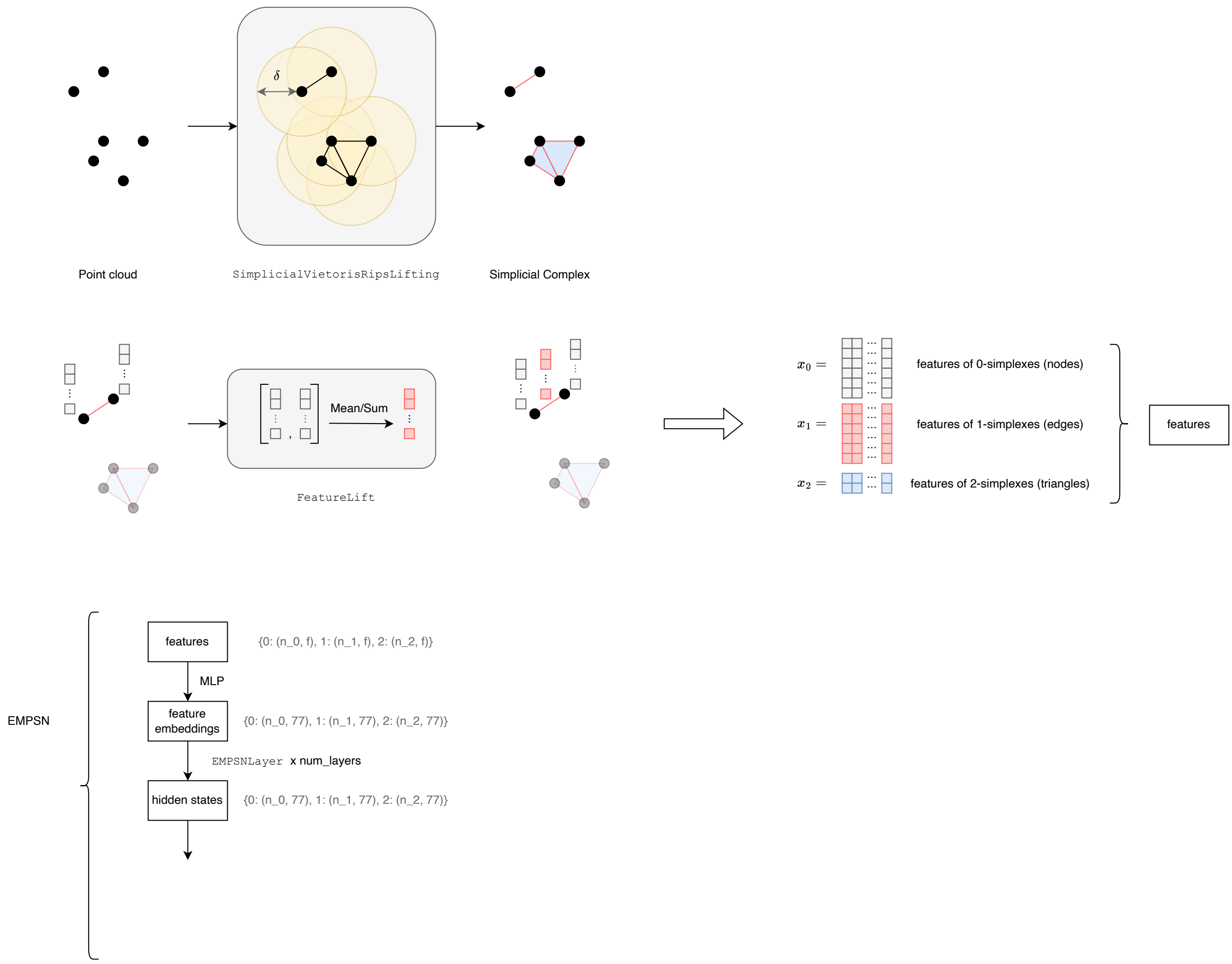

Figure 1: collection of explanatory images

3 strengths and weaknesses

An immediate strength of the proposed graph neural network is the utilization of the invariant geometric framework, combined with the topological information that is generated through the lifting operation. This leads to a very data-efficient learning environment with comparable results to state-of-the-art (SotA) methods. This strength is evident in the method's performance on the QM9 dataset. Even though the current method is general and does not utilize a large network, it is comparable to, or even outperforms, more specific methods (that utilize domain-specific strategies) on certain features.

Another strength is the general applicability of this method. The proposed equivariant simplicial message passing network (EMPSN) is theoretically applicable to any dataset, since we perform the lifting operation. However, data that has a certain structure is likely to utilize its effects better. Additionally, it scales more efficiently to data (it scales based on the simplicial structures and their depth).

Finally, as mentioned before, it appears empirically that this method performs well even when the higher-order simplicial structures aren't being utilized much. It also effectively prevents over-smoothing, a common problem in graph neural networks (GNNs).

The current implementation also has several practical and theoretical weaknesses.

Practically, the readability of the code is very poor without explanation and contains some random hardcoded parts that should not be there, notwithstanding the bugs that were found. There are illogical design choices using strings. There is a lot of experimentation going on with different approaches.

The original paper describes two experiments: one using an Invariant MPSN and one that uses the full Equivariant MPSN. Unfortunately, the code for the latter isn't included in the paper, so there is no way to reproduce those results. The former hardcoded the potential invariances (e.g., from rank 0 to 1, there are 3), but there is no explanation of what this number is based on, nor is it clear how this number increases into higher-level simplicial structures, i.e., what would be the rank 2 to 3 invariances? The paper lacks justification for many design choices in this regard.

Another aspect that isn't clear or explained is the 'search range' (Δ) that the Vietoris-Rips algorithm uses.

Moreover, the results in the original paper seem to be too good. Reproduction attempts with the original code show very similar trends, but they don't seem to score as high as the original paper implies. This could be a coincidence, as it hasn't been tested over several seeds due to computational cost.

A design choice that isn't clear is how 'high' one should go with the simplicial structures and their communication. For the current paper, it was chosen that the communication between simplices was only over the same rank and rank +1. As an exception, however, 2nd-order simplices are not allowed to communicate amongst each other. It is unclear if this choice is grounded in empirical results, nor is it clear what the 'ideal' height should be of the lifting and corresponding message passing. One could, for example, allow communication not only upward but also downward, or go to the 4-dimensional level and see if that garners better results. All of these choices are likely also dataset-dependent.

4 Describe your novel contribution.

For our contribution, we decided to attempt to reformat the code to adhere to the new TopoX framework. TopoX is a topological deep learning suite built on top of pytorch and it is meant to create a general framework and provide building blocks for geometric and topological deep learning applications. The current paper is a prime candidate to utilize this framework. Besides, TopoX creators have placed a request for people to translate lifting operations into their framework. Both these additions are qualitatively useful additions to not only the current paper, streamlining and clarifying the procedures, but also the TopoX suite, by increasing code written by their standards.

5 Results of your work

Our code is located in this forked repository: https://github.com/martin-carrasco/challenge-icml-2024.git

5.1 As a summary:

- We have reformatted the code for the Vietoris-Rips lift.
- We have attempted to restructure the codebase as much as possible, using the building blocks of TopoX, specifically the ones for simplicial layers and the message passing framework. By doing so we have changed the input required for the layers from the indexing-based structure that the original authors used, into a adjacency/incidence matrix format, which is a more generalizable structure.

- The main changes are in the lifting subfolder and in the models subfolder.

6 Conclusion

In this post we have investigated a novel 'proof of concept' study that investigates the use of simplicial structures combined with geometric properties as a useful avenue to study. the paper designs an equivariant or invariant message passing simplicial network. It seems that the author is fair in his assessment of the utility. It is a compact network and the usage of invariance combined with simplices seems prudent, although the accuracy of the predictions seems to not reach the same level as the original paper. We name a couple of found strengths and weaknesses, focussing on the efficiency and generalisability. While also remarking the difficulty of the code and its implementation. By refitting the code into a promising new suite for topological deep learning, named TopoX, we hope to make the code more accessible and to contribute to the growing amount of papers using TopoX.

7 Contributions

- Nordin Belkacemi - focused on the code and theory
- Andreas Berentzen - wrote the lifting and focused on the invariance understanding, wrote intro for the blogpost
- Martin Carrasco - main codewriter, implemented the lifting code and the EMPSN neural network
- Valeria Sepicacchi - wrote the theoretical background and helped with EMPSN code
- Jesse Wonnink - wrote rest of the blogpost and helped with EMPSN code