On the expressive power of message-passing neural networks as global feature map transformers*

Floris Geerts[1] [0000−0002−8967−2473], Jasper Steegmans[2] [0000−0003−2087−9430], and Jan Van den Bussche[2] [0000−0003−0072−3252]

[1] University of Antwerp
[2] Hasselt University

An important issue in machine learning is the choice of formalism to represent the functions to be learned [9,10]. For example, feedforward neural networks with hidden layers are a popular formalism for representing functions from $\mathbb{R}^n$ to $\mathbb{R}^p$. When considering functions over graphs, graph neural networks (GNNs) have come to the fore [8]. GNNs come in many variants; in this paper, specifically, we will work with the variant known as message-passing neural networks (MPNNs) [3].

MPNNs compute numerical values on the nodes of an input graph, where, initially, the nodes already store vectors of numerical values, known as *features*. Such an assignment of features to nodes may be referred to as a *feature map* on the graph [5]. We can thus view an MPNN as representing a function that maps a graph, together with a feature map, to a new feature map on that graph. We refer to such functions as *global feature map transformers (GFMTs)*.

Of course, MPNNs are not intended to be directly specified by human designers, but rather to be learned automatically from input–output examples. Still, MPNNs do form a language for GFMTs. Thus the question naturally arises: what is the expressive power of this language?

We believe GFMTs provide a suitable basis for investigating this question rigorously. The G for ‘global’ here is borrowed from the terminology of *global function* introduced by Gurevich [6,7]. Gurevich was interested in defining functions in structures (over some fixed vocabulary) *uniformly*, over all input structures. Likewise, here we are interested in expressing GFMTs uniformly over *all* input graphs. We also consider infinite subclasses of all graphs, notably, the class of all graphs with a fixed bound on the degree.

As a concrete handle on our question about the expressive power of MPNNs, in this paper we define the language MPLang. This language serves as a yardstick for expressing GFMTs, in analogy to the way Codd’s relational algebra serves as a yardstick for relational database queries [1]. Expressions in MPLang can define features built arbitrarily from the input features using three basic operations also found in MPNNs:

1. Summing a feature over all neighbors in the graph, which provides the message-passing aspect;

* Originally appeared in Proceedings FoIKS 2022 (LNCS vol. 13388).

2. Applying an activation function, which can be an arbitrary continuous function;
3. Performing arbitrary affine transformations (built using constants, addition, and scalar multiplication).

The difference between MPLang-expressions and MPNNs is that the latter must apply the above three operations in a rigid order, whereas the operations can be combined arbitrarily in MPLang. In particular, every MPNN is readily expressible in MPLang. If the converse also holds, which we will investigate, then MPLang would serve as a benchmark for GFMTs that can be expressed by combining the operations of an MPNN in a free manner.

Our research question can thus be made concrete: is, conversely, every GFMT expressible in MPLang also expressible by an MPNN? We offer the following answers.

1. Considering the case of the popular activation function ReLU [4,2]. In this case, we show that for every MPLang expression there exists an equivalent MPNN and show to generate such an MPNN.
2. When arbitrary activation functions are allowed, we show that the above still holds when restricting the input to any class of graphs of bounded degree, equipped with features taken from a bounded domain. We also again show how to generate these MPNNs.
3. Finally, when the MPNN is required to use only the ReLU activation function except of in the last layer, we show that every MPLang expression can still be *approximated* by such an MPNN. For this result we again restrict to graphs of bounded degree, and moreover to features taken from a bounded domain. Here the conversion from MPLang to MPNN invokes the classical universality of feedforward neural networks.
   Additionally, we show that if at least one of these restrictions does not hold, that such an approximation does not exist.

For our proofs we require the construction of complex MPLang expressions and complex MPNNs. To aid in their construction, we define several operations on GFMTs. We then prove that as long as MPNNs are closed under these operations and the translation of the building blocks of MPLang can be expressed as MPNNs, then any MPLang expression can be converted into an MPNN. As a result of this modular approach, if we wish to prove any formalism expressing GFMTs can be converted into another formalism expressing a GFMT, we simply need to prove these conditions hold for these two formalisms.

References

1. Abiteboul, S., Hull, R., Vianu, V.: Foundations of Databases. Addison-Wesley (1995)
2. Arora, R., Basu, A., Mianjy, P., Mukherjee, A.: Understanding deep neural networks with rectified linear units. In: Proceedings 6th International Conference on Learning Representations. OpenReview.net (2018)

3. Gilmer, J., Schoenholz, S., et al.: Neural message passing for quantum chemistry. In: Precup, D., Teh, Y. (eds.) Proceedings 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1263–1272 (2017)
4. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
5. Grohe, M.: The logic of graph neural networks. In: Proceedings 36th Annual ACM/IEEE Symposium on Logic in Computer Science. pp. 1–17. IEEE (2021)
6. Gurevich, Y.: Algebras of feasible functions. In: Proceedings 24th Symposium on Foundations of Computer Science. pp. 210–214. IEEE Computer Society (1983)
7. Gurevich, Y.: Logic and the challenge of computer science. In: Börger, E. (ed.) Current Trends in Theoretical Computer Science, pp. 1–57. Computer Science Press (1988)
8. Hamilton, W.: Graph Representation Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool (2020)
9. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson, fourth edn. (2022)
10. Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press (2014)