

# *Time2Vec and the Nuances of Temporal Representation*

*Me et al.*

*July 22, 2023*

Time2Vec, as described in the paper by Borealis et al.<sup>1</sup>, offers a vector representation of time, aiming to capture the characteristics of time in machine learning models. However, it can be susceptible to misinterpretation in various domains. This essay tries to explore the potential pitfalls of Time2Vec, especially when juxtaposed against the transformer architecture.

<sup>1</sup> Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupard, and Marcus Brubaker. Time2vec: Learning a vector representation of time, 2019

## *Strengths Revisited*

1. Time2Vec's ability to encapsulate the progression, periodicity, and scale of time is of great interest. The representation, while being versatile across datasets, captures time's essence in a manner that's more sophisticated than mere timestamps.
2. The transformer architecture, though originally crafted for NLP, has been employed in numerical timeseries classification, as seen in various articles (show articles). Time2Vec's potential as a positional encoding mechanism in such architectures underscores its versatility.

## *Concerns and Clarifications*

1. Many articles seem to use timeseries values (like prices or sales) as inputs to Time2Vec, rather than the actual temporal information (i.e., a time index  $t = 0, 1, 2, 3, \dots$ ). This approach seems more like an enrichment of feature representation rather than true positional encoding. Even the original Word2Vec paper<sup>2</sup>, which inspired Time2Vec, hints at its design being more aligned with time-related features.
2. Terms like "positional encoding" have specific meanings. If Time2Vec is used to encode features rather than positions, it leads to potential confusion. Such confusion can compound when architectures are layered and interwoven in complex models.
3. Even if Time2Vec were applied on a time index, it should likely be a global time index created before batching or windowing the data. This is because the local time index within a batch might be arbitrary, potentially introducing inconsistencies from a theoretical point of view.

<sup>2</sup> Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013

## *Conclusion*

Time2Vec, in its essence, presents an alternative approach to representing time. However, its application needs careful consideration. Misinterpretations, especially in areas as intricate as positional encoding, can lead to flawed models and misguided conclusions. A more rigorous mathematical framework and clearer guidelines for Time2Vec’s application might be required.

## *Acknowledgements*

This report was typeset using L<sup>A</sup>T<sub>E</sub>X, originally developed by Leslie Lamport and based on Donald Knuth’s T<sub>E</sub>X. A template that can be used to format documents with this look and feel has been released under the permissive APACHE LICENSE 2.0, and can be found online at <https://tufte-latex.github.io/tufte-latex/>.

## *References*

Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus Brubaker. Time2vec: Learning a vector representation of time, 2019.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.