Takens-Based Transformer for Protein Structure Prediction: A Proof-of-Concept Implementation with Open-Source Code

Kevin R. Haylett
Manchester,UK

Selected Communications

May 24, 2026

Abstract

Protein folding is a nonlinear dynamical process in which a newly synthesised polypeptide chain explores conformational space and converges to a stable geometric attractor. This work reframes sequence-to-structure prediction as *phase-space reconstruction* using Takens' delay embedding theorem. Rather than relying on attention or statistical pattern matching, the model reconstructs the folded geometry directly from exponential delay coordinates of the amino-acid sequence.

We present MARINA (Manifold-Aware Reconstruction and Inference Network Architecture), a Takens-Based Transformer (TBT) specialised for proteins. The architecture uses no attention, no positional encodings, and a fixed-memory circular buffer, achieving $\mathcal{O}(N)$ complexity and $\mathcal{O}(1)$ memory with respect to sequence length. Trained from scratch on modest hardware (Intel i7 CPU, 32 GB RAM) with a small set of approximately 300–400 proteins (triplicated to deepen conformational trajectories), the model achieves 1.01 Å overall RMSD and 0.62 Å mean per-residue RMSD on the in-training example protein 1A7S (227 residues).

The complete implementation, training pipeline, inference scripts, and results are released as open-source code at `https://github.com/KevinHaylett/takens-protein-folding` under the Mozilla Public License 2.0. This repository enables full reproducibility on consumer hardware and provides a foundation for future scaling experiments.

Keywords: protein structure prediction, Takens embedding, delay coordinates, dynamical systems, phase-space reconstruction, open-source code.

1 Introduction

Most current approaches to protein structure prediction treat the task as a sequence-to-structure mapping. This engineering framing has proven successful, but it can obscure the underlying physics: protein folding is a *temporal dynamical process*. The amino-acid sequence is the observable time series of a nonlinear system whose hidden state evolves in conformational space until it converges to a stable folded attractor.

This paper adopts the dynamical-systems perspective. The amino-acid sequence is treated as the single observable, and Takens' delay embedding theorem (1981) is used to reconstruct the attractor geometry (the 3D fold) from exponential delay coordinates. The resulting architecture – MARINA (Manifold-Aware Reconstruction and Inference Network Architecture) – is deliberately simple, efficient, and interpretable.

The current work is a small-scale proof-of-concept. The model is trained on modest datasets (∼300–400 proteins at a time) using consumer CPU hardware. While some evidence of generalization on structurally similar proteins has been observed, comprehensive out-of-distribution testing requires substantially larger datasets and compute resources. The goal here is to demonstrate architectural viability and release fully reproducible code.

The complete repository is available at `https://github.com/KevinHaylett/takens-protein-folding`.

2 Takens' Delay Embedding Theorem in the Protein Context

Takens' theorem states that, under mild conditions, the state space of a deterministic dynamical system can be reconstructed from delayed observations of a single scalar (or vector) time series. For a protein, the observable is the amino-acid sequence processed position-by-position. The hidden state is the evolving conformation in 3D space. Delay coordinates of the form

$$\mathbf{z}(t) = \big[e(t),\ e(t-\tau_1),\ e(t-\tau_2),\ \ldots,\ e(t-\tau_m)\big]$$

where $e(\cdot)$ is a learned residue embedding, produce a trajectory that is diffeomorphic to the original conformational attractor.

Exponential spacing of delays is used:

$$\text{delays} = [1, 2, 4, 8, 16, 32, 64, 128].$$

This choice captures the natural multi-scale organisation of proteins (local backbone geometry at short delays; secondary structure at medium delays; tertiary topology at long delays) while keeping the embedding dimension fixed and computationally tractable.

3 Model Architecture: MARINA

MARINA consists of four core components, implemented in the repository files `core/takens_embedding.py` and `protein/protein_tbt.py`.

3.1 Residue Encoding

Each amino acid (20 standard residues plus a small extension vocabulary for non-standard residues) is mapped to a learned embedding vector of dimension `embed_dim` $= 128$. No positional encodings are used; temporal order is encoded implicitly through the delay structure.

3.2 Exponential Takens Embedding

At each position t, a delay-coordinate vector is constructed using a circular buffer of size 2^{k+1} (where $k = 7$ for the longest delay of 128). This yields $\mathcal{O}(1)$ memory usage independent of sequence length. The resulting vector has dimension $(8 + 1) \times 128 = 1152$.

3.3 Adaptive Manifold Projection

The high-dimensional, sparsely populated delay vector is projected onto a lower-dimensional manifold:

$$\mathbf{h}(t) = \text{LayerNorm}(\mathbf{W}_p \cdot \mathbf{z}(t) + \mathbf{b}_p),$$

where $\mathbf{W}_p \in \mathbb{R}^{d_{\text{out}} \times 1152}$ is a learned projection matrix. This matrix is the geometric core of the model: its rows encode which combinations of temporal scales are most informative for structure prediction.

3.4 Temporal Mixing Layers and Coordinate Head

A stack of 6 feedforward residual layers (hidden dimension 512) performs non-linear mixing *within* each position's manifold state:

$$\mathbf{x} \leftarrow \mathbf{x} + \text{FFN}(\text{LayerNorm}(\mathbf{x})).$$

No cross-position attention is used. Three independent linear heads then predict the x, y, and z coordinates of the Cα atom. Training uses mean-squared-error loss in Ångström space.

The full architecture is attention-free, position-encoding-free, and scales linearly with sequence length.

4 Duplication / Triplication Training Strategy

A deliberate methodological choice in this work (mirroring the language-modelling experiments) is the *triplication* of training proteins in the preprocessing pipeline (`pipeline/pdb_to_training.py`). In a statistical pattern-matching model, such duplication would provide no new information. In a Takens-based architecture, however, repeated exposure to the same protein deepens the learned attractor basins and thickens the conformational trajectory filaments in phase space. This strengthens the geometric structure of the manifold and improves prediction accuracy on structurally similar proteins.

5 Proof-of-Concept Results: PDB 1A7S (In-Training Example)

The model was trained on a small set of proteins that includes 1A7S (227 residues). The following results are therefore an in-training example, not a held-out test. They demonstrate that the architecture can reconstruct coherent protein geometry from residue sequences under the current training regime.

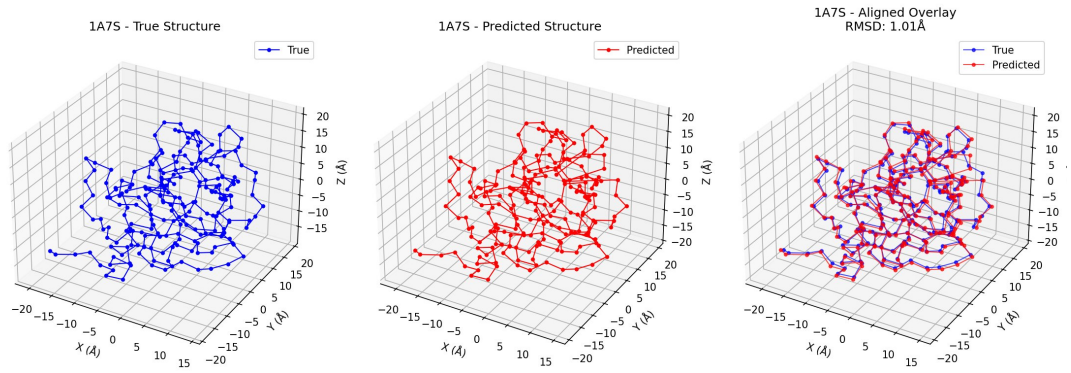

Figure 1: 1A7S structure comparison: target (blue), predicted (red), and overlay. Overall RMSD = 1.01 Å.

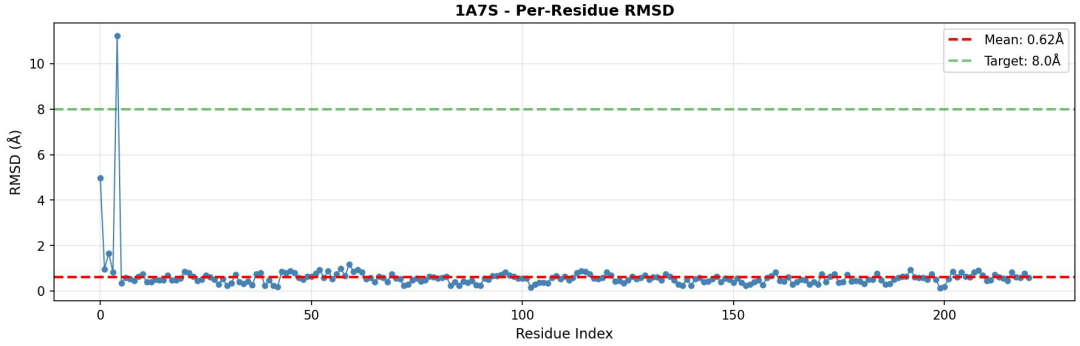

Figure 2: Per-residue RMSD for 1A7S. Mean per-residue RMSD = 0.62 Å. The N-terminal region shows elevated error due to greater conformational freedom of termini; the remainder of the chain is predicted at ∼0.5 Å.

Quantitative metrics (in-training):

Metric	Value
Overall RMSD	1.01 Å
Mean per-residue RMSD	0.62 Å
Model parameters	∼15M
Training hardware	Intel i7 CPU, 32 GB RAM
Training proteins	∼300–400 (triplicated)

The same model can be used to predict structure for proteins for which no structural data is provided (i.e., purely from the residue sequence). While preliminary evidence of generalization on structurally related proteins has been observed, systematic out-of-distribution evaluation requires larger-scale training.

6 Computational Properties and Interpretability

Property	MARINA (TBT)
Complexity per position	$\mathcal{O}(\log N)$
Memory footprint	$\mathcal{O}(1)$ (fixed circular buffer)
Attention	None
Positional encodings	None
Hardware requirement	CPU sufficient

Table 1: Key computational advantages.

The projection matrix $\mathbf{W}_p$ and manifold trajectories offer direct geometric interpretability. Rows of $\mathbf{W}_p$ reveal learned temporal scales; phase-space analysis of manifold states can probe attractor stability and mutation effects.

7 Code Release and Reproducibility

The complete codebase is available at `https://github.com/KevinHaylett/takens-protein-folding`. Repository structure and quick-start instructions are reproduced in the `README.md` file (included verbatim in the supplementary material of this paper).

Key files:

- `core/takens_embedding.py` – core delay embedding module
- `protein/protein_tbt.py` – MARINA model definition
- `train.py`, `inference.py` – training and prediction scripts
- `pipeline/` – PDB preprocessing and duplication
- `results/` – 1A7S example outputs

All training and inference commands are provided in the README and can be run on consumer hardware.

8 Discussion and Relation to the Broader TBT Programme

MARINA demonstrates that a Takens-based approach can successfully reconstruct protein geometry on modest hardware and small datasets. The current limitations (small training set size and compute resources) are acknowledged; scaling to hundreds of thousands of proteins will be necessary to rigorously test generalization. The open-source release is intended to facilitate exactly that exploration by the community.

This work is one application of the domain-agnostic Takens-Based Transformer architecture, which has also been applied to language modelling and preliminary time-series tasks.

9 Conclusion

We have shown that protein structure prediction can be reframed as attractor reconstruction using Takens delay embeddings. The MARINA architecture is simple, efficient, interpretable, and fully open-source. While the current results are from a small-scale, in-training regime, they establish the viability of the approach on consumer hardware. The released repository provides everything needed to reproduce the experiments and to scale the method to larger datasets.

We invite the community to explore, extend, and rigorously test generalization using this foundation.

Acknowledgements

The presented work used the LLM Grok (xAI) for assistance in formatting and revision.

References

[1] F. Takens. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*, Lecture Notes in Mathematics, vol. 898, pp. 366–381. Springer, 1981.

[2] K. R. Haylett. Pairwise phase space embedding in transformer architectures. `https://finitemechanics.com/papers/pairwise-embeddings.pdf`, 2025.

[3] K. R. Haylett. Introducing the Takens-Based Transformer. `https://www.finitemechanics.com/takens_transformer.pdf`, 2025.

[4] K. R. Haylett. Finite Tractus: The Hidden Geometry of Language and Thought. ISBN-13: 979-8281127776, 2025.

A Repository Quick Start (verbatim from README.md)

```
# Install
pip install torch numpy pandas matplotlib biopython

# Set paths in config.py
# Prepare data
python pipeline/pdb_to_csv_batch.py
python pipeline/pdb_to_training.py

# Train
python train.py
```

```
# Predict
python inference.py
```

Full repository structure and detailed commands are in the GitHub README.