PRIMA: a bidirectional state-space architecture and training approach for sequence modelling of protein-protein interactions

Arturo Fiorellini-Bernardis [1] **Sebastien Boyer** [1] **Jean Quentin** [1] **Ghassene Jebali** [1] **Oliver Bent** [1]

Abstract

Protein-protein interactions (PPIs) govern essential biological processes, yet building protein language models (PLMs) that directly capture the complexity of interacting multi-chain sequences remains an open challenge. Transformer-based PLMs rely on self-attention whose quadratic complexity makes processing long interacting complexes resource-intensive, leaving an emerging gap between a rich diversity of current single-chain PLMs and scalable PPI PLMs. We present PRIMA, a proof-of-concept PLM for PPIs built on bidirectional Mamba (BiMamba), a selective state-space model that scales linearly with sequence length and processes arbitrarily long inputs without positional encodings. PRIMA is trained in two phases: first on diverse single protein sequences to learn general representations, then on concatenated PPI pairs to capture interaction-specific context. At an 8 million parameter scale and equal count to a transformer baseline, PRIMA (i) outperforms transformer-based PLMs on standard-length inputs, (ii) outperforms them on longer sequences and length extrapolation tasks, and (iii) does so at significantly lower computational cost, establishing BiMamba as a strong backbone for scalable PPI PLMs.

1. Introduction

Protein-protein interactions (PPIs) are central to immune response, signal transduction, metabolic control, and gene regulation, and their study is key to understanding disease mechanisms and guiding drug discovery (Greenblatt et al., 2024; Nada et al., 2024). Having a deep learning system able to model PPIs efficiently is an actively researched topic (Lee, 2023) that presents two fundamental problems: the lack of large, well curated PPI datasets (Kovtun et al., 2024), and the computational cost that comes with the size of interacting proteins complexes (Celik & Xie, 2024). Sequence-based protein language models (PLMs) are a natural vehicle for learning PPI representations: experimentally acquired sequences are far more abundant and less noisy than structures, and by solving a masked language modeling (MLM) task on large corpora a model implicitly learns evolutionary, structural, and functional constraints (Elnaggar et al., 2021; Lin et al., 2022). This reasoning extends naturally to interacting sequences: training on concatenated PPI pairs exposes the model to the joint statistics of the complex, allowing it to learn interaction-relevant patterns without experimental labels.

The vast majority of SOTA PLMs are transformer-based: from autoregressive models that excel at generation (Nijkamp et al., 2022), to encoder-only architectures that outperforms them on representation and downstream tasks (Elnaggar et al., 2021). The self-attention mechanism scales quadratically with sequence length ($O(L^2)$), making training and inference a challenge for multi-chain complexes that routinely span thousands of amino acids. Typically, a practical cap of 1024 tokens is set during training. Models with fixed positional encodings (PEs) (e.g. ESM-1 (Rives et al., 2019)) cannot process longer inputs at all; models with rotary PE (RoPE) (e.g. ESM-2 (Lin et al., 2022)) degrade outside the training distribution. As a result, there is a significant gap between the rich landscape of single-protein PLMs and scalable foundation models for PPIs.

We introduce PRIMA, a proof-of-concept foundation PLM for PPIs built on bidirectional Mamba (BiMamba) (Gu & Dao, 2024). Being a selective state-space model (SSM), PRIMA scales linearly with the input length ($O(L)$), is inherently position-aware thanks to its recurrent nature and requires no positional encodings, conferring natural length extrapolation. We train PRIMA with self-supervised masked language modeling (MLM) in two phases: we pre-train the model on diverse proteins from UniRef50 (The UniProt Consortium, 2023) to learn general protein representations; then, we post-train it on a large corpus of concatenated PPI pairs from STRING (Szklarczyk et al., 2023), so that it

[1]InstaDeep Ltd, London, The United Kingdom. Correspondence to: Arturo Fiorellini-Bernardis <a.fiorellini@instadeep.com>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026.

learns interaction-specific context. In this proof-of-concept study, we train an ∼8M parameter model on subsets of UniRef50 and STRING and evaluate it on different single sequence and PPI benchmarks. We compare PRIMA to an encoder-only transformer baseline with the ESM-2 8M architecture and similar parameter count, trained with the same protocol on the same data; we refer to this model as Protein Encoding Transformer (PET). We show that PRIMA outperforms PET on standard-length and longer inputs, as well as on length extrapolation tasks. Moreover, PRIMA achieves this at a fraction of the computational cost, owing to the linear rather than quadratic scaling of SSMs with sequence length.

2. Related Work

The works most directly related to PRIMA are MINT (Ullanat et al., 2026), LC-PLM (Wang et al., 2024), and ProtMamba (Sgarbossa et al., 2024). MINT is a transformer-based foundation PLM for PPIs: it encodes each chain independently with ESM-2 650M, then models inter-chain dependencies via cross-attention, training on a curated STRING split with MLM. Due to the quadratic cost of attention, MINT caps input length at 512 tokens per chain, randomly cropping longer sequences. LC-PLM introduces the same BiMamba-S building block as PRIMA, trained on UniRef50 with MLM and extended (as LC-PLM-G) to multi-protein sequences via random walks on a protein interaction graph, improving performance on remote homology and PPI link prediction. However, LC-PLM-G is never trained on actual concatenated protein complexes and is not evaluated on binding affinity prediction. ProtMamba is a causal Mamba model trained on MSA-concatenated sequences with a fill-in-the-middle objective, targeting within-family tasks such as mutational effect prediction and protein generation; it is not designed for cross-protein interaction modeling. More details can be found in Appendix A. None of these works trains a bidirectional SSM directly on concatenated interacting protein sequences and evaluates on PPI representation quality and binding affinity prediction: PRIMA addresses precisely this gap.

3. Methods

We introduce PRIMA, a PLM for PPIs built with bidirectional Mamba layers. The goal of PRIMA is to generate the best PPI representation possible: to achieve this, we train the model with a MLM objective in two distinct phases, single sequences and PPIs. As mentioned in Section 1 PRIMA is tested against PET, trained on SOTA transformer-based PLM architectures.

3.1. Architecture

PRIMA is a stack of N bidirectional Mamba (BiMamba-S) blocks, a token embedding layer, and a tied LM head. We choose a bidirectional architecture because it consistently outperforms causal models on representation tasks (Devlin et al., 2019), and because causal Mamba would yield an asymmetric PPI representation: positions in chain A could never attend to chain B, while B sees all of A. The bidirectional design doubles the constant factor but preserves $O(L)$ scaling.

Input sequences are tokenized at the amino acid level (vocabulary of 29 tokens: 20 standard amino acids, 4 non-standard and ambiguous residues, and 5 special tokens). PPI pairs are structured as `<cls>`+[**A**]+`<inter>`+`<cls>`+[**B**]. Positional encodings are not used: the information is implicitly encoded in the SSM recurrent state, which confers natural length extrapolation (Gu & Dao, 2024; Wang et al., 2024).

Each BiMamba-S block runs two independent Mamba passes (forward and reverse) with shared input/output projections but direction-specific SSM parameters, summing their outputs before a residual connection (see Appendix B for a detailed description). The LM head is a linear projection tied to the token embedding matrix, reducing parameter count and regularising the output representation. We build PRIMA at small scale with 7.3M parameters (N=16 layers, D=256). For direct comparison, PET uses the ESM-2 8M architecture (7.8M parameters, N=6 layers, D=320).

3.2. Data

PRIMA and PET are trained, validated, and tested on the same data (full details are given in Appendix C). We cap sequence length at 1000 amino acids (AA) during training to match the typical maximum input length of transformer-based PLMs (ESM-2 caps at 1024 tokens), enabling a fair comparison and allowing us to test length extrapolation.

UniRef50 is a non-redundant protein sequence database clustering UniProtKB at 50% sequence identity, retaining one representative per cluster. It is the standard pre-training corpus for PLMs (ESM-2, LC-PLM) due to its scale, diversity, and low redundancy. We filter sequences to lengths 50–1000 AA and draw a random subsample of 5M sequences, holding out 5% for validation (UniRef50_5M).

STRING (v12) is a database of known and predicted PPIs covering physical binding and functional associations across thousands of organisms. We start from the train/validation split used by MINT (96M training pairs, 250k validation pairs). From the training split, we filter to pairs where both sequences satisfy $200 \leq L \leq 1000$ AA, then apply stratified degree-weighted node sampling to select ∼500k representative sequences, retaining only edges where both endpoints are selected and deduplicating bidirectional pairs.

This yields 12M PPI pairs for training (STRING_12M). From the validation split, applying the same length filter yields 112k pairs used during validation.

3.3. Training Strategy

PRIMA and PET are trained on a single 80GB GPU (NVIDIA H100) with a MLM objective, which has been shown effective in order to learn to encode different biophysical properties, such as thermal stability or hydrophobicity (Elnaggar et al., 2021; Lin et al., 2022). Specifically, the masking follows the BERT protocol, with a 15% of the input tokens masked. The training of PRIMA and PET is done in 2 phases: a first pre-training phase on single sequences to learn general protein representations and statistics, a second post-training phase on PPIs to learn interaction specific information. During the first phase, the models are pretrained for 50 epochs on UniRef50_5M to learn general protein representations from a distribution spanning all biological domains. During the second phase, the models are post-trained for 8 epochs on STRING_12M PPI pairs. The sequences constituting a PPI pair are concatenated before being fed to the models, as described in Section 3.1. For a significant portion of the PPI pairs, the length of the concatenated input spans thousands of AA, making training PET burdensome on a single GPU. Moreover, during the pretraining phase of PET, RoPE are trained up to 1000 AA, similar to ESM-2. For these reasons, when a PPI pair exceeds this threshold, the input is centered around the `<inter>` token and cropped to the maximum length used during training.

3.4. Testing

3.4.1. Zero-shot Perplexity

After training, we test PRIMA and PET on held-out sets from STRING (refer to Appendix C for more details). After phase 1, we test both models on STRING_TEST_SS: 236k individual sequences spanning lengths 8–32,000 AA. After phase 2, we test on STRING_TEST_PPI: 138k PPI pairs where at least one sequence falls outside the training length distribution. In both cases the task mirrors training (15% of tokens are masked and perplexity (defined in Appendix E.0.1) is measured on the masked positions) testing both in-distribution performance and length extrapolation without any post-training.

3.4.2. Binding Prediction Benchmarks

We adapt the benchmarking pipeline from MINT (Ullanat et al., 2026). We test on four datasets: HumanPPI (Xu et al., 2022) (binary PPI classification, ∼27k pairs, accuracy); GoldStandardPPI (Bernett et al., 2024) (leakage-free binary PPI classification, ∼274k pairs, AUPRC); YeastPPI (Xu et al., 2022) (binary PPI classification, ∼4.5k pairs, accuracy); and SKEMPI 2.0 (Jankauskaitė et al., 2019) ($\Delta\Delta G$ regression on point mutations across 345 complexes, ∼7k datapoints, Pearson correlation, complex-level cross-validation). Rather than encoding chains separately as in MINT, we feed concatenated PPI pairs as a single input to both models, as explained in Section 3.1. A shallow MLP probe (details in Appendix D) is trained on frozen embeddings extracted via mean pooling over residues (special tokens excluded); any improvement over equal embedding sizes reflects representation quality rather than probe capacity. Note that PET has an embeddings size of $D = 320$, which is 25% larger than PRIMA's $D = 256$, resulting in a bigger downstream MLP probe (+ ∼25%) that gives PET an advantage. Results are reported for both cropped inputs (following the MINT protocol, capping at 512 tokens per chain) and full-length inputs (PRIMA processes these natively; PET is capped at 18,000 AA per pair to avoid OOM on a single H100). Refer to Appendix C for more details.

3.4.3. Efficiency and Scaling Benchmark

We benchmark forward-pass latency and peak GPU memory as a function of sequence length ($L \in 128, 256, 512, 1024, 2048, 4096, 8192, 16384$) and batch size ($B \in 1, 8, 32$) on a single NVIDIA H100 80GB GPU, using synthetic polyalanine inputs to isolate compute cost. Median latency is computed over 50 timed passes after 5 warmup passes, with `torch.cuda.synchronize()` ensuring accurate GPU-side timing.

4. Results

4.0.1. Zero-shot Benchmarks

PRIMA outperforms PET across the entire length distribution on both STRING_TEST_SS and STRING_TEST_PPI (Figures 3–4 in Appendix E). The gap increases monotonically beyond the training distribution, reaching 21.7% lower perplexity on single sequences and 18.7% on PPI pairs for inputs longer than 5000 AA. PET's degradation at long lengths is driven by RoPE operating outside its training regime, compounded in the PPI setting by the cropping applied during post-training.

4.0.2. Binding Prediction Benchmarks

Figure 1 shows results on the four binding prediction benchmarks. PRIMA outperforms PET on all tasks, both with cropped and full-length inputs. Providing full-length inputs improves performance for both models, but the gain is larger for PRIMA, which processes complete PPI context without truncation. Note that PRIMA's advantage is conservative, as PET's embedding dimension (D=320) is 25% larger than PRIMA's (D=256), giving PET a larger downstream MLP probe.

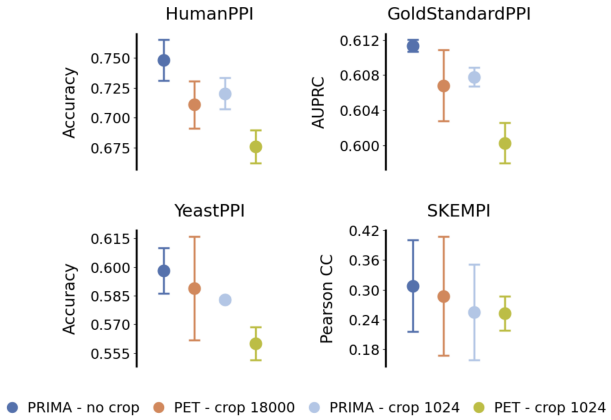

Figure 1. Performance of PRIMA and PET on binding prediction benchmarks. The models are fed concatenated PPIs to extract the embeddings on which an MLP probe is trained. The models are fed both cropped sequences (512 tokens per single sequences before concatenation), and full sequences (PET's input is limited to 18000 AAs to avoid OOM errors on one NVIDIA H100 80 GB GPU).

4.0.3. Efficiency

Table 1 reports latency and memory at batch size 1 for different lengths. At $L \leq 1024$, PET is faster due to highly optimised attention kernels; the crossover occurs around $L \approx 2048$. Beyond this, PRIMA scales linearly while PET scales quadratically, PRIMA reaching $11.5\times$ lower latency and $115\times$ lower memory at L=16384. At larger batch sizes PET runs out of memory for $L \geq 4096$; PRIMA processes all tested lengths without OOM (full results in Appendix E).

Table 1. Forward-pass latency (ms per sequence) and peak GPU memory (GB) for PRIMA and PET at batch size 1 on an NVIDIA H100 80 GB GPU. Speedup $= t_{\text{PET}}/t_{\text{PRIMA}}$; values > 1 indicate PRIMA is faster.

	ms/seq			Mem (GB)	
Length	PRIMA	PET	Speedup	PRIMA	PET
128	14.3	3.4	0.24×	0.07	0.07
256	14.3	3.5	0.24×	0.07	0.08
512	15.0	3.4	0.23×	0.07	0.11
1024	14.3	5.3	0.37×	0.08	0.25
2048	14.4	12.6	0.87×	0.10	0.76
4096	14.6	33.7	2.30×	0.14	2.80
8192	18.1	105.4	5.83×	0.22	10.91
16384	33.4	383.9	11.5×	0.37	43.23

5. Conclusions and Outlooks

This work introduces PRIMA, a PLM approach for learning PPIs, designed to validate the architecture and training strategy prior to further scaling. The goal of PRIMA is to generate information dense representations of PPIs that encode the interaction context directly. PRIMA is built on bidirectional Mamba layers, which scale linearly $O(L)$ with the input length as opposed to the vast majority of SOTA PLMs such as ESM-2, built on vanilla transformer layers that scale quadratically $O(L^2)$. Based on our knowledge, PRIMA is the first SSM-based PLM for PPIs.

PRIMA has 7.3M parameters, and undergoes two distinct training stages. During the first stage, the model is pre-trained on a subset of UniRef50 to learn single protein general representations. During the second stage, the model is post-trained on concatenated PPIs coming from a subset of STRING, a collection of PPIs. To compare PRIMA to SOTA models with similar parameter count, we use the same protocol to train PET, an attention-based PLM with the ESM-2 8M architecture. PRIMA is tested against PET in different benchmarks: zero-shot perplexity and length extrapolation on single sequences after the first training phase; zero-shot perplexity and length extrapolation on PPIs after the second training phase; SOTA binding affinity benchmark with regression tasks using a shallow MLP probe trained on the embeddings; computational cost in terms of latency and memory. PRIMA shows better performance than PET in all the benchmarks while being faster (up to $11.5\times$) and more memory efficient (up to $115\times$). PRIMA is built on the first `mamba-ssm` code release: we are exploring newer and more efficient implementations that promise to push the performance even further.

We believe PRIMA presents a few key advantages for PPI modeling. It processes inputs of arbitrary length, while scaling linearly. It handles length extrapolation naturally, as positional information is implicitly encoded in the SSM recurrent state. Also, residue identity at position i is predominantly constrained by local context ($\sim$±10 residues): secondary structure elements, Ramachandran dihedral preferences, and BLOSUM evolutionary constraints all operate over short windows, making the SSM state's exponential decay over sequence distance a useful inductive bias for single-chain modeling. At the same time, the ability to propagate information across the full concatenated PPI sequence is critical for capturing inter-chain co-evolutionary signals: interface residues in chain A are co-constrained by residues in chain B that may be hundreds of positions away in the concatenated input. The selective mechanism in Mamba allows the model to operate at both scales simultaneously.

The success of PRIMA at prototype scale is encouraging: the architecture and training strategy validate cleanly, and the primary bottleneck is now scale rather than design. We intend to scale both the model size and training corpus, using the full UniRef50 and STRING databases. Moreover, we plan to investigate more fundamental questions about what representation learning on concatenated PPIs actually captures: whether the model learns genuine inter-chain co-evolutionary signals at the interface, or primarily richer intra-chain statistics; and whether the learned representations encode binding affinity independently of single-chain

thermodynamic stability, quantities that single-sequence PLMs cannot disentangle by construction.

References

Alanis-Lobato, G., Andrade-Navarro, M. A., and Schaefer, M. H. HIPPIE v2.0: enhancing meaningfulness and reliability of protein–protein interaction networks. *Nucleic Acids Research*, 45(D1):D408–D414, 2017. doi: 10.1093/nar/gkw985.

Bernett, J., Blumenthal, D. B., and List, M. Cracking the black box of deep sequence-based protein–protein interaction prediction. *Briefings in Bioinformatics*, 25(2): bbae076, 2024. doi: 10.1093/bib/bbae076.

Celik, M. H. and Xie, X. Efficient inference, training, and fine-tuning of protein language models. *bioRxiv*, 2024. doi: 10.1101/2024.10.22.619563. URL https://www.biorxiv.org/content/early/2024/10/25/2024.10.22.619563.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2019. doi: 10.18653/v1/N19-1423.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Yu, W., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., and Rost, B. Prottrans: Towards cracking the language of lifes code through self-supervised deep learning and high performance computing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021. doi: 10.1109/TPAMI.2021.3095381.

Greenblatt, J. F., Alberts, B. M., and Krogan, N. J. Discovery and significance of protein-protein interactions in health and disease. *Cell*, 187(23):6501–6517, 2024. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2024.10.038. URL https://www.sciencedirect.com/science/article/pii/S0092867424012534.

Gu, A. and Dao, T. Mamba: Linear-time sequence modeling with selective state spaces. 2024. URL https://arxiv.org/abs/2312.00752.

Jankauskaitė, J., Jiménez-García, B., Dapkūnas, J., Fernández-Recio, J., and Moal, I. H. SKEMPI 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. *Bioinformatics*, 35(3):462–469, 2019. doi: 10.1093/bioinformatics/bty635.

Kovtun, D., Akdel, M., Goncearenco, A., Zhou, G., Holt, G., Baugher, D., Lin, D., Adeshina, Y., Castiglione, T., Wang, X., Marquet, C., McPartlon, M., Geffner, T., Corso, G., Stärk, H., Carpenter, Z., Kucukbenli, E., Bronstein, M., and Naef, L. Pinder: The protein interaction dataset and evaluation resource. *bioRxiv*, 2024. doi: 10.1101/2024.07.17.603980. URL https://www.biorxiv.org/content/early/2024/07/20/2024.07.17.603980.

Lee, M. Recent advances in deep learning for protein-protein interaction analysis: A comprehensive review. *Molecules*, 28(13), 2023. ISSN 1420-3049. doi: 10.3390/molecules28135169. URL https://www.mdpi.com/1420-3049/28/13/5169.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*, 2022.

Nada, H., Choi, Y., Kim, S., Jeong, K. S., Meanwell, N., and Lee, K. New insights into protein–protein interaction modulators in drug discovery and therapeutic advance. *Signal Transduction and Targeted Therapy*, 2024.

Nijkamp, E., Ruffolo, J., Weinstein, E. N., Naik, N., and Madani, A. Progen2: Exploring the boundaries of protein language models. *arXiv*, 2022.

Peri, S., Navarro, J. D., Amanchy, R., Kristiansen, T. Z., Jonnalagadda, C. K., Surendranath, V., Niranjan, V., Muthusamy, B., Gandhi, T. K. B., Gronborg, M., et al. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome Research*, 13(10):2363–2371, 2003. doi: 10.1101/gr.1680803.

Press, O. and Wolf, L. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 157–163, 2017. doi: 10.18653/v1/E17-2025.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., and Fergus, R. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *PNAS*, 2019. doi: 10.1101/622803. URL https://www.biorxiv.org/content/10.1101/622803v4.

Sgarbossa, D., Malbranke, C., and Bitbol, A. F. ProtMamba: a homology-aware but alignment-free protein state space model. *bioRxiv*, 2024. doi: 10.1101/2024.05.24.595730. URL https://www.biorxiv.org/content/early/2024/05/25/2024.05.24.595730.

Szklarczyk, D., Kirsch, R., Koutrouli, M., Nastou, K., Mehryary, F., Hachilif, R., Gable, A. L., Fang, T., Doncheva, N. T., Pyysalo, S., Jensen, L. J., and von Mering, C. The STRING database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic Acids Research*, 51(D1):D638–D646, 2023. doi: 10.1093/nar/gkac1000.

The UniProt Consortium. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1): D523–D531, 2023. doi: 10.1093/nar/gkac1052.

Ullanat, V., Jing, B., Sledzieski, S., and Berger, B. Learning the language of protein-protein interactions. *Nature Communications*, 2026.

Wang, Y., Wang, Z., Sadeh, G., Zancato, L., Achille, A., Karypis, G., and Rangwala, H. Long-context protein language model, 2024. URL https://arxiv.org/abs/2411.08909.

Xu, M., Zhang, Z., Lu, J., Zhu, Z., Zhang, Y., Ma, C., Liu, R., and Tang, J. PEER: A comprehensive and multi-task benchmark for protein sequence understanding. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

A. Related Work

MINT is a transformer-based PLM for PPIs. It uses ESM-2 650M to encode each protein chain independently, and then adds cross-attention layers to model inter-chain dependencies. The model is trained on a curated version of STRING with an MLM objective, and it is evaluated on PPI tasks including binding affinity and mutational effect prediction, outperforming single-chain PLMs such as ESM-2 alone. Due to the quadratic cost of attention, MINT caps input length at 512 tokens per chain, randomly cropping longer sequences.

LC-PLM is a bidirectional Mamba-based PLM (BiMamba-S, with shared projection layers) trained on individual sequences from UniRef50 with MLM, demonstrating superior length extrapolation compared to transformer-based PLMs. In a second training stage, LC-PLM is extended to LC-PLM-G, which continues training on multi-protein sequences constructed from random walks on graphs of interacting proteins from ogbn-proteins (Open Graph Benchmark). Individual chains are separated by special tokens encoding edge type, `<edge>` and `<no_edge>`, giving the model explicit supervision over interaction topology. LC-PLM-G improves downstream performance on remote homology prediction, protein function prediction (ogbn-proteins), and PPI link prediction. However, the model is never trained on actual concatenated protein complexes and is not evaluated on binding affinity prediction.

ProtMamba is a Mamba-based model, which unlike PRIMA and LC-PLM adopts a causal (unidirectional) architecture trained on concatenated sequences from the same multiple sequence alignment (MSA), making it homology-aware without requiring expensive explicit alignments. It is trained using a fill-in-the-middle (FIM) objective on the OpenProteinSet (16M MSAs) and can process up to 2^{17} tokens at 107M parameters. ProtMamba focuses on within-family tasks: mutational effect prediction, enzyme activity prediction, and protein generation/inpainting. It is not designed for cross-protein interaction modeling and is not evaluated on PPI-specific tasks.

B. PRIMA Architecture

PRIMA is built as a stack of N BiMamba blocks, a token embedding layer, and a tied language model (LM) head.

Token embedding and special tokens. Input sequences are tokenized at the amino acid level using a vocabulary of 29 tokens covering the 20 standard amino acids, 4 non-standard and ambiguous residues, and 5 special tokens. The `<inter>` token is inserted as a separator between the two chains of a PPI pair, i.e. inputs are structured as `<cls>` $+ [\mathbf{A}] +$ `<inter>` $+$ `<cls>` $+ [\mathbf{B}]$. Token embeddings are learned via $\mathbf{E} \in \mathbb{R}^{V \times D}$ (V is the vocabulary size, D the embedding dimension) with no positional encodings: positional information is implicitly encoded in the recurrent state of the SSM, which confers natural length extrapolation capability.

BiMamba-S block. Each block takes an input $\mathbf{T}_{n-1} \in \mathbb{R}^{B \times L \times D}$ and produces $\mathbf{T}_n \in \mathbb{R}^{B \times L \times D}$ (B is batch size, L is sequence length) via a residual connection:

$$\mathbf{T}_n = \text{BiMamba}(\text{Norm}(\mathbf{T}_{n-1})) + \mathbf{T}_{n-1}$$

where Norm is RMSNorm applied before the mixer (pre-norm). The BiMamba mixer runs two independent Mamba passes in parallel, one on the original sequence and one on its reverse, and sums their outputs. Each directional pass follows the standard selective SSM block: the input is linearly projected to two branches $\mathbf{X}$ and $\mathbf{Z}$ (expanded dimension $E = 2D$); $\mathbf{X}$ passes through a 1D depthwise convolution and SiLU activation before the SSM scan, while $\mathbf{Z}$ serves as a multiplicative gate on the SSM output: $\mathbf{Y}' = \text{SSM}(\mathbf{X}) \odot \text{SiLU}(\mathbf{Z})$. The gated outputs of the forward and reverse passes are summed and projected back to dimension D via the output projection $\mathbf{W}_{\text{out}}$. Formally:

$$\text{BiMamba}(\mathbf{T}) = \mathbf{W}_{\text{out}} \left(\mathbf{Y}'_{\text{fwd}} + \mathbf{Y}'_{\text{rev}}\right)$$

where $\mathbf{Y}'_{\text{fwd}}$ and $\mathbf{Y}'_{\text{rev}}$ are the gated SSM outputs of the forward and reverse passes respectively.

Shared projection layers (BiMamba-S). Following LC-PLM, the input projection $\mathbf{W}_{\text{in}}$ (mapping to $\mathbf{X}$ and $\mathbf{Z}$) and the output projection are shared between the forward and reverse passes. The selective SSM parameters (state transition matrices $\mathbf{A}$, input projections $\mathbf{B}$, $\mathbf{C}$, and timescale $\mathbf{\Delta}$) remain direction-specific, preserving independent dynamics for each scan direction. This design, denoted BiMamba-S, reduces the parameter cost of the projection layers relative to having fully separate forward and reverse modules, enabling deeper models at the same parameter budget. LC-PLM reports a 4.5% improvement in evaluation loss from BiMamba-S over the unshared variant at matched parameter count.

LM head. A linear projection $\mathbf{W}_{\text{LM}} \in \mathbb{R}^{D \times V}$ maps the final hidden states to logit scores over the vocabulary. Its weights are tied to the token embedding matrix ($\mathbf{W}_{\text{LM}} = \mathbf{E}^\top$), reducing the parameter count and regularising the output representation

(Press & Wolf, 2017). Note that LC-PLM reports marginal gains with untied weights; we retain tying for parameter efficiency.

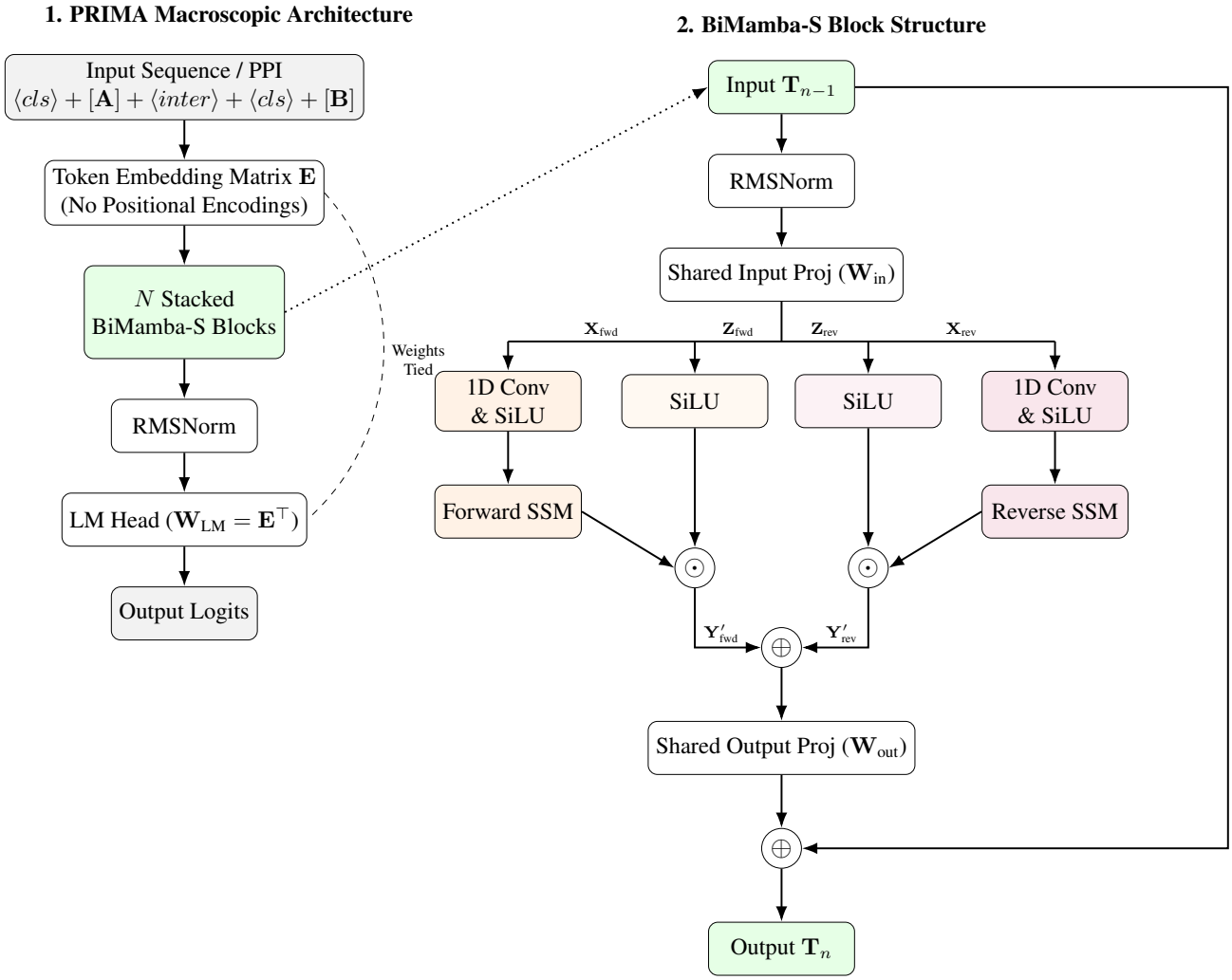

Figure 2. Comprehensive Architecture of PRIMA. (1) Macroscopic pipeline showing the embedding layer, lack of positional encodings, and tied LM head. (2) Microscopic BiMamba-S Block showing the shared projections and the specific $\mathbf{X}$ and $\mathbf{Z}$ gating branches where $\mathbf{Y}' = \text{SSM}(\mathbf{X}) \odot \text{SiLU}(\mathbf{Z})$.

In the interest of efficiency, we build a small scale model with 7.3 million parameters, which can be directly compared to PET, built with the ESM-2 8M architecture (7.8M parameters). For PRIMA this results in $N = 16$ BiMamba layers, and an embedding size $D = 256$. PET has $N = 6$ self-attention layers and an embedding dimension $D = 320$.

C. Data

C.0.1. Training and Validation data

UniRef50 (UniProt Reference Clusters at 50% identity) is a non-redundant protein sequence database produced by the UniProt Consortium. It clusters all sequences in UniProtKB (plus selected UniParc sequences) such that any two sequences within the same cluster share at least 50% sequence identity over 80% of the shorter sequence. Only the representative sequence of each cluster is retained, dramatically reducing redundancy while preserving diversity. As of recent releases it contains ∼60M representative sequences spanning all domains of life. It is the standard pre-training corpus for protein language models (ESM-2, LC-PLM, etc.) precisely because it is large, diverse, and non-redundant at a level that prevents the model from simply memorizing near-identical sequences. We pre-filter UniRef50 retaining sequences of length between

50 and 1000 AA: this is aligned with the maximum input length used by ESM-2 and allows testing length extrapolation capabilities in both directions after training. Then we take a random subsample of 5M sequences, of which we hold-out a random 5% for validation: UniRef50_5M.

STRING (Search Tool for the Retrieval of Interacting Genes/Proteins) is a database of known and predicted PPIs, covering both physical binding and functional associations (co-expression, genomic context, text mining, transferred homology); the training corpus therefore captures broad co-functional relationships in addition to direct binding partners. The full database (v12) contains hundreds of millions of interaction records across thousands of organisms. We start from the training and validation split done by the authors of MINT, which retained 96M PPI pairs for training, and 250k pairs for validation. The sequence length ranges from 8 to 32,000 AA. From this, we subsample the training split in the following way: we pre-filter the 96M PPIs by retaining pairs where both sequences must satisfy $200 \leq L \leq 1000$ AA. Pairs where either partner falls outside this range are discarded. Then we perform node selection via stratified degree-weighted sampling, aiming to select 500,000 unique sequences (nodes) from the length-filtered PPI graph. Sequences are first binned into 15 log-spaced length bins between 200 and 1000 AA (these thresholds are chosen to test the length extrapolation abilities of the models during the evaluation). Within each bin, sequences are sampled with probability proportional to degree$^{\alpha}$ with $\alpha = 0.9$, i.e. with a strong preference for high-degree hub proteins (well-connected proteins are more likely to be retained). The per-bin sample quota is proportional to the bin's fraction of total nodes. This is done in order to bias the subset to present strong PPI signals. After selecting $\sim$500,000 nodes, only edges where both endpoints are in the selected set are kept. Bidirectional duplicates (A,B) / (B,A) are deduplicated by keeping only the lexicographically canonical form. This results in a subset of 12M deduplicated PPI pairs: STRING_12M. Regarding the validation dataset, we pre-filter the 250,000 PPIs and retain pairs where both sequences must satisfy $200 \leq L \leq 1000$ AA. Pairs where either partner falls outside this range are discarded. We end up with 112k PPI pairs.

STRING_TEST We refer to the 138k PPI pairs held out from the validation set as STRING_TEST_PPI; in this split, at least one of the sequences' length is out of the training distribution. From STRING_TEST_PPI, we extract all the 236k individual sequences forming the PPI pairs, with length spanning from 8 to 32,000 AA: we refer to this set as STRING_TEST_SS.

C.0.2. Benchmark Data

HumanPPI (Xu et al., 2022) – A binary PPI classification dataset for human proteins from the PEER benchmark. Positive pairs are experimentally confirmed interactions sourced from (Peri et al., 2003); negatives are constructed from proteins in different subcellular compartments, making non-interaction biologically plausible. The train/validation/test split follows the PEER protocol: within-split redundancy is first removed at 90% sequence identity, then a 40% pairwise sequence identity cutoff is applied between each pair of splits to reduce leakage. The task is binary classification (interacts / does not interact), evaluated by accuracy. Importantly, (Bernett et al., 2024) showed that even benchmarks using 40% cutoff splits remain susceptible to node-degree bias, models can achieve inflated accuracy by learning hub-protein statistics rather than genuine interaction features, making performance on this dataset an optimistic upper bound on true generalisation. The dataset contains $\sim$27,000 datapoints.

GoldStandardPPI (Bernett et al., 2024) – A leakage-free binary PPI classification benchmark for human proteins introduced to address a widespread data leakage problem in prior PPI prediction benchmarks, where sequence similarity between train and test splits allowed models to exploit node-degree shortcuts rather than learning genuine interaction features. Positive interactions are sourced from HIPPIE v2.3 (Alanis-Lobato et al., 2017), with experimentally validated human PPIs; negatives are randomly sampled protein pairs with degree preserved in expectation. The three-way train/validation/test split (Intra1/Intra0/Intra2, $\sim$163k/59k/52k pairs) is constructed using graph partitioning on the sequence similarity graph followed by a 40% pairwise sequence identity cutoff between splits, ensuring no protein appears in more than one split and that all splits are internally non-redundant. The task is binary classification (interacts / does not interact); the primary evaluation metric is AUPRC. Prior state-of-the-art methods achieve AUPRC ≈ 0.59 on this benchmark, as reported in the paper. The dataset contains $\sim$274,000 datapoints.

YeastPPI (Xu et al., 2022) – A binary PPI classification dataset for S. cerevisiae proteins. Positive pairs are experimentally confirmed yeast PPIs; negatives are constructed from proteins in different subcellular compartments, making non-interaction biologically plausible. The train/validation/test split follows the PEER protocol: within-split redundancy is first removed at 90% sequence identity, then a 40% pairwise sequence identity cutoff is applied between each pair of splits to reduce leakage. The task is binary classification (interacts / does not interact), evaluated by accuracy. (Bernett et al., 2024) specifically re-evaluated yeast PPI datasets under leakage-free conditions and found that all tested deep learning models drop

to near-random performance, suggesting that reported accuracies on this dataset largely reflect data leakage and node-degree exploitation rather than learned interaction features. YeastPPI should therefore be interpreted as a relatively noisy signal; strong performance reflects representational quality but weak performance does not necessarily imply failure of the model. The dataset contains ∼4,500 datapoints.

SKEMPI 2.0 (Jankauskaitė et al., 2019) – A curated structural database of experimentally measured changes in binding free energy ($\Delta\Delta G$) upon point mutation, covering mutations across 345 protein complexes. The task is regression: predicting $\Delta\Delta G$ for a mutated PPI. Evaluation uses Pearson correlation between predicted and measured $\Delta\Delta G$. The train/test split is done by protein complex (3-fold cross-validation), so the model is tested on complexes it has never seen. Only dimers are retained, and the dataset contains ∼7,000 datapoints.

D. Binding Prediction Benchmark MLP Probe

For HumanPPI and GoldStandardPPI the MLP has a single hidden layer of dimension 512 and ReLU activation, a single output unit, and a dropout probability of 0.2, the optimizer is AdamW, learning rate of 1e-3 with adaptive scheduling, it is trained for 50 epochs, and results are averaged over 3 runs. YeastPPI and SKEMPI are smaller datasets, and the probe architecture is adapted to prevent overfitting: the MLP has a single hidden layer of dimension 64 and ReLU activation, a single output unit, and a dropout probability of 0.2, the optimizer is AdamW, learning rate of 1e-3 with adaptive scheduling, it is trained for 100 epochs, and results are averaged over 3 runs.

E. Detailed Results

E.0.1. Masked Language Modeling Perplexity

In Section 3.4.1, we evaluate the zero-shot representation quality of PRIMA and PET using perplexity. Because both models are trained with a Masked Language Modeling (MLM) objective, the evaluation mirrors the training task, and perplexity is computed specifically over the masked positions.

Given an input tokenised sequence S, M then denotes the set of indices corresponding to the masked tokens (15% of the input), and $|M|$ the total number of masked positions. The masked sequence fed to the model is denoted as $S_{\setminus M}$. The MLM perplexity (PPL_{MLM}) is defined as the exponential of the average negative log-likelihood over the masked tokens:

$$\text{PPL}_{\text{MLM}}(S) = \exp\left(-\frac{1}{|M|}\sum_{i\in M}\log P(s_i \mid S_{\setminus M})\right) \tag{1}$$

where s_i represents the ground-truth sequence token at position i, and $P(s_i \mid S_{\setminus M})$ is the probability assigned by the model's language modeling head to the correct token given the masked sequence.

E.0.2. Zero-shot Benchmarks

Figure 3 shows the perplexity of the PRIMA and PET on STRING_TEST_SS; Figure 4 shows the perplexity of the PRIMA and PET on STRING_TEST_PPI. Results are binned by length. PRIMA outperforms PET across the entire length distribution, with a gap that increases monotonically for sequences longer than the training distribution. For very long inputs, longer than 5000 AA, the perplexity of PRIMA is 21.7% lower for single sequences, 18.7% for PPIs. The degradation in PET's performance over long sequences is given by RoPE that are not trained in this regime, and thus inject noise in the model. The performance of both models decreases on short sequences. The performance gap between the two models is greater for PPIs than single sequences in the range $200 \leq L \leq 2000$ AA: this could be due to the fact that the perplexity is overall lower (both models were trained longer and on more data, possibly closer to the distribution of the test set).

E.0.3. Binding Prediction Benchmarks

Figure 1 shows the performance of PRIMA and PET over the MINT benchmarks. Results are shown for cropped and uncropped inputs; note that PET can process inputs only up to 18,000 AA with batch size 1 to prevent OOM errors on one NVIDIA H100 80GB GPU: this limit has an effect only on the GoldStandardPPI benchmark. The results with cropped sequences follow this protocol: in case a PPI exceeds 1024, the individual sequences are randomly cropped to 512 minus special tokens and then concatenated. PRIMA outperforms PET in all the benchmarks, while being significantly more

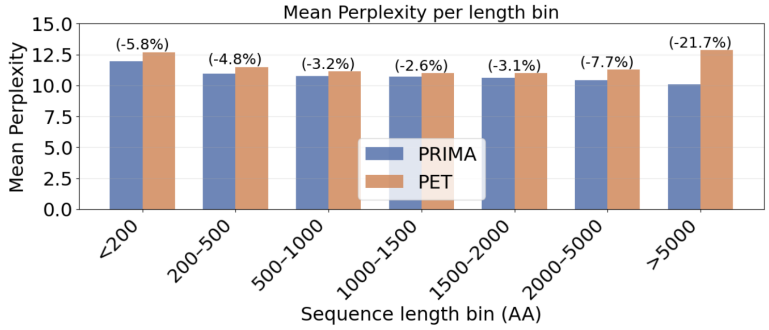

Figure 3. Mean perplexity of PRIMA and PET on single sequences from STRING_TEST_SS binned by input length. The blue bars correspond to PRIMA, the orange bars correspond to PET. Both models are tested after the first phase of training on the subset UniRef50_5M.

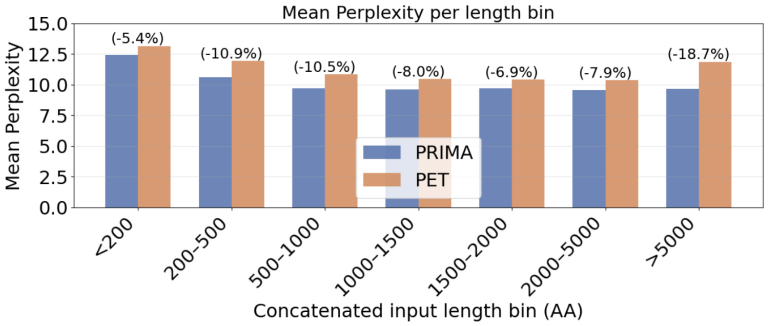

Figure 4. Perplexity of PRIMA and PET on PPIs from STRING_TEST_PPI binned by input length. The blue bars correspond to PRIMA, the orange bars correspond to PET. Both models are tested after the second phase of training on the subset STRING_12M.

efficient, both in terms of speed and memory. Letting the model have access to the entire PPI avoiding any cropping proves to be beneficial for both PRIMA and PET.

E.0.4. Efficiency and Scaling Benchmark

At short to moderate lengths ($L \leq 1024$) for batch size 1, PET is roughly 3–4$\times$ faster per sequence: the quadratic attention cost is negligible for short inputs and the ESM-2 implementation benefits from highly optimised attention kernels. The crossover occurs around $L \approx 2048$ at batch size 1 ($\sim$14.4 ms for PRIMA vs $\sim$12.6 ms for PET), and around $L \approx 1024$ at larger batch sizes.

Beyond the crossover, PRIMA scales linearly while PET scales quadratically. At $L = 4096$, PRIMA is 2.3$\times$ faster than PET at batch size 1 (Table 1; at $L = 8192$, 5.8$\times$ faster; at $L = 16384$, 11.5$\times$ faster (33 ms vs 384 ms per sequence). For batch size 8, PET runs out of memory for $L = 8192$ (Table 2; for batch size 32, OOM occurs already at $L = 4096$ (Table 3. PRIMA processes all tested lengths at all batch sizes without OOM.

The memory advantage is more dramatic than the latency advantage, directly reflecting the $O(N)$ vs $O(N^2)$ scaling. At $L = 1024$, PRIMA uses $\sim$84 MB vs PET's $\sim$246 MB (3$\times$ less). At $L = 4096$, the ratio is 142 MB vs 2804 MB ($\sim$20$\times$). At $L = 16384$, PRIMA requires 375 MB while PET requires 43 GB at batch size 1, a 115$\times$ difference, and OOMs entirely at batch sizes ≥ 8.

Notably, both PRIMA's and PET's per-sequence latency is nearly constant from at small batch sizes and small lengths: the fixed CUDA kernel overhead dominates over the actual compute. In this regime, PET is faster due to better optimized kernels of the HuggingFace implementation. PRIMA's throughput saturates around 500k–550k tokens/sec at large lengths and batch sizes, suggesting it is compute-bound in this regime. For PPI inputs, where concatenated sequences routinely span 500–2000+ amino acids, PRIMA operates in the regime where it is already competitive with or faster than PET, while consuming a fraction of the memory.

Table 2. Forward-pass latency (ms per sequence) and peak GPU memory (GB) for PRIMA and PET at batch size 8 on an NVIDIA H100 80 GB GPU.

	MS/SEQ			MEM (GB)	
LENGTH	PRIMA	PET	SPEEDUP	PRIMA	PET
128	1.94	0.43	0.22×	0.08	0.10
256	1.83	0.45	0.25×	0.10	0.17
512	1.86	0.86	0.46×	0.14	0.45
1024	2.42	4.04	1.67×	0.22	1.51
2048	4.51	11.17	2.48×	0.38	5.64
4096	8.18	31.86	3.90×	0.68	21.94
8192	15.89	OOM	—	1.30	OOM
16384	30.70	OOM	—	2.53	OOM

Table 3. Forward-pass latency (ms per sequence) and peak GPU memory (GB) for PRIMA and PET at batch size 32 on an NVIDIA H100 80 GB GPU.

	MS/SEQ			MEM (GB)	
LENGTH	PRIMA	PET	SPEEDUP	PRIMA	PET
128	0.46	0.15	0.33×	0.14	0.20
256	0.64	0.30	0.47×	0.22	0.50
512	1.12	0.72	0.64×	0.38	1.61
1024	2.24	3.89	1.74×	0.68	5.83
2048	4.35	10.90	2.51×	1.30	22.34
4096	8.20	OOM	—	2.53	OOM
8192	15.61	OOM	—	5.00	OOM
16384	29.87	OOM	—	9.92	OOM