FAST PROTEOME-SCALE PROTEIN INTERACTION RETRIEVAL VIA RESIDUE-LEVEL FACTORIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Protein-protein interactions (PPIs) are mediated at the residue level. Most sequence-based PPI models consider residue-residue interactions across two proteins, which can yield accurate interaction scores but are too slow to scale. At proteome scale, identifying candidate PPIs requires evaluating nearly *all possible protein pairs*. For N proteins of average length L, exhaustive all-against-all search requires $\mathcal{O}(N^2L^2)$ computation, rendering conventional approaches computationally impractical. We introduce RaftPPI, a scalable framework that approximates residue-level PPI modeling while enabling efficient large-scale retrieval. RaftPPI represents residue interactions with a Gaussian kernel, approximated efficiently via structured random Fourier features, and applies a low-rank factorized attention mechanism that admits pooling into a compact embedding per protein. Each protein is encoded once into an indexable embedding, allowing approximate nearest-neighbor search to replace exhaustive pairwise scoring, reducing proteome-wide retrieval from *months* to *minutes* on a single GPU. On the human proteome with the D-SCRIPT dataset, RaftPPI retrieves the top 20% candidate pairs ($\sim$200M) in 6 GPU minutes, covering 75.1% of the true interacting pairs, compared to 4.9 GPU months for the best prior method (61.2%). Across seven benchmarks with sequence- and degree-controlled splits, RaftPPI achieves state-of-the-art PPI classification and retrieval performance, while enabling residue-aware, retrieval-friendly screening at proteome scale.

1 INTRODUCTION

Protein-protein interaction (PPI) is central to understanding cellular mechanisms and enabling applications in target discovery (Loscalzo, 2023), pathway reconstruction (Ritz et al., 2016), and functional annotation (Sharan et al., 2007). In practice, many discovery tasks require proteome-scale screening, scoring protein pairs within a species to surface plausible interactors (Humphreys et al., 2024; Zhang et al., 2024). However, proteome-scale PPI prediction remains time-consuming due to the quadratic number of candidate protein pairs. One high-accuracy route is to predict multimer structures with end-to-end structure predictors (Jumper et al., 2021; 2024; Evans et al., 2021). While accurate, these pipelines rely on MSAs/templates and require $\mathcal{O}(L^3)$ complexity to update pairwise representations of proteins with length L. This daunting cost makes per-complex structure prediction difficult to amortize across large candidate sets. An alternative frames PPI as binary classification from alignment-free PLM embeddings (Sledzieski et al., 2024; Ko et al., 2024; Liu et al., 2024). Although per-sequence encoding is $\mathcal{O}(L^2)$, these models must jointly encode each protein *pair* at inference; thus an exhaustive screen over the human proteome with $\sim$ 20,000 proteins entails $\approx 2\times10^8$ candidate pairs, making large-scale screening computationally prohibitive. To illustrate, the state-of-the-art PPI classification model PLM-Interact (Liu et al., 2024) requires 148.47 A100 GPU-days (≈4.9 months) to screen the human proteome (see Table 3 and § 4.3).

In light of these limitations, we propose **R**esidue-interaction **A**pproximation with **F**ourier Fea**T**ures (RaftPPI), which models PPI by *approximating residue-level interactions while enabling scalable protein retrieval*. RaftPPI models residue–residue scores with a Gaussian kernel and aggregates them to a protein-level score (Fig. 1). The non-linear kernel is efficiently approximated via random Fourier features (Rahimi and Recht, 2007; Yu et al., 2016), and pooling uses a low-rank factorized attention that admits a linear-time approximation at inference. As a result, each protein is encoded once into a fixed-length representation amenable to approximate nearest-neighbor search, e.g., Hierarchical navigable small world (HNSW (Malkov and Yashunin, 2020)), to retrieve likely interactors—retaining

residue-level interactions while avoiding explicit per-pair computation. In practice, the dominant cost is PLM encoding, giving overall $\mathcal{O}(NL^2)$ across a proteome; on a single A100, retrieving the top 20% of human-proteome pairs completes in minutes, achieving a 10^4-fold speedup over prior art.

Besides model design, we also seek to mitigate challenges from the lack of reliable negative data in PPI datasets. Experimentally confirmed non-interactions are rare, so negatives are typically *constructed*, and their quality varies widely (Neumann et al., 2022; Zhao et al., 2022). Random or compartment-based pairing often produces overly easy, biased examples that lead to overly optimistic results, whereas co-localized, functionally related, or topology-aware sampling yields harder and more informative ones (Ben-Hur and Noble, 2006; Park and Marcotte, 2011; 2012; Zhang et al., 2018). These observations motivate our *adaptive negative weighting* loss, which applies self-adversarial weights (Sun et al., 2019) based on model confidence so that harder negatives are assigned greater weights, mitigating biases from easy constructed pairs.

Contributions. We introduce RaftPPI, a retrieval-friendly, residue-aware framework that compresses each protein into an indexable embedding for efficient ANN search, reducing proteome-level PPI screening from *GPU months* to *minutes*. With an adaptive negative-weighting loss that emphasizes hard negatives and mitigates the lack of reliable negative data, RaftPPI achieves strong classification and retrieval performance under rigorous sequence- and degree-controlled benchmarks.

2 Related Work

Protein Language Models. Transformer-based Protein Language Models (PLMs) pretrained on large sequence corpora (e.g., ProtTrans (Elnaggar et al., 2021), ESM 1b (Rives et al., 2021), and ESM 2 (Lin et al., 2023)) learn residue-level embeddings that implicitly capture evolutionary and structural priors. Generative PLMs such as ProGen and ProGen2 (Madani et al., 2020; Nijkamp et al., 2022) use autoregressive modeling for controllable sequence design. Beyond sequence-only vocabularies, Foldseek (van Kempen et al., 2024) introduces discrete 3D structure tokens that have been used to augment PLMs with structure-aware vocabularies (SaProt (Su et al., 2023)) or to predict structure tokens directly (ISM (Ouyang-Zhang et al., 2025)). These models provide transferable features that support many modern PPI predictors, including our method.

Protein–Protein Interaction Prediction. An ideal way to assess PPIs is to predict the 3D structure of protein complexes directly using end-to-end structure predictors such as AlphaFold2 (Jumper et al., 2021), RoseTTAFold (Baek et al., 2021), AlphaFold-Multimer (Evans et al., 2021), and the more recent AlphaFold3 (Jumper et al., 2024). These systems achieve remarkable accuracy but are computationally intensive, often rely on multiple sequence alignments (MSAs) and/or structural templates, and require explicit per-pair inference. Another line of work formulates PPI as a graph machine learning problem (e.g., link prediction (Nasiri et al., 2021)) on residue- or protein-level graphs. Examples include diffusion-state methods (Devkota et al., 2020) and GNN-based approaches such as GNN-PPI (Lv et al., 2021), SGPPI (Huang et al., 2023), PPI-GNN (Jha et al., 2022), and HIGH-PPI (Gao et al., 2023). Although graph priors can be powerful, the performance of such models depends on the availability and quality of the underlying network and can be vulnerable to degree bias and data leakage (Bernett et al., 2024). These limitations prevent proteome-level screening. In light of this, we focus on sequence-only methods, as they are alignment-free (no MSA needed) and graph-free, while maintaining good performance.

Among sequence-only methods, early work such as SPRINT (Li and Ilie, 2017) computes pair-specific similarities with spaced-seed k-mer hashing. Subsequent deep learning encoders include DeepFE-PPI (Yao et al., 2019), the fully connected and LSTM models of Richoux et al. (Richoux et al., 2019), and PIPR's Siamese residual RCNN (Chen et al., 2019). Many later sequence models follow the D-SCRIPT (Sledzieski et al., 2021) paradigm, computing residue–residue interaction scores and aggregating them into a protein-level score; Topsy-Turvy (Singh et al., 2022) and TT3D (Sledzieski et al., 2023) further extend this approach by incorporating graph priors (Devkota et al., 2020) and structural embeddings (van Kempen et al., 2024). Recently, PLM-based approaches (Sledzieski et al., 2024; Ko et al., 2024; Yang et al., 2024; Liu et al., 2024) leverage the strong performance of pretrained PLMs (Lin et al., 2023; Elnaggar et al., 2021) to model interactions at residue-level resolution. These approaches are typically accurate but *non-factorizable*: they jointly encode each protein pair, which makes proteome-scale retrieval computationally prohibitive.

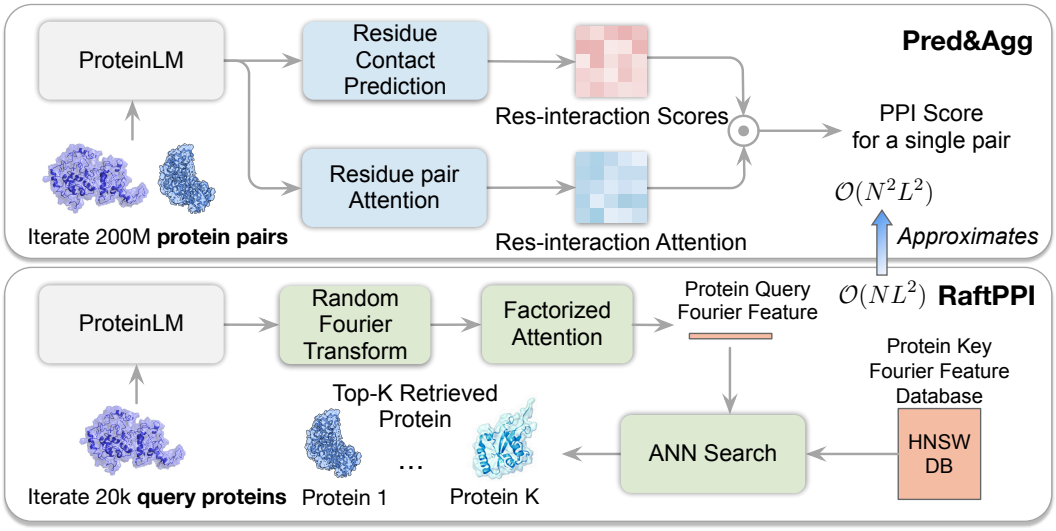

Figure 1: **Overview of RaftPPI.** RaftPPI approximates a standard PPI pipeline *Pred&Agg* (above) that *predicts* residue-level contact scores and *aggregates* them using attention to a final PPI score. The *Pred&Agg* pipeline costs $\mathcal{O}(N^2L^2)$ for proteome-wide retrieval. RaftPPI uses Random Fourier Features to approximate a Gaussian kernel and leverages separable attention. The pipeline is factorizable (below), enabling per-protein Fourier-feature embeddings and ultra-fast retrieval via approximate nearest neighbor search (e.g., HNSW) at proteome scale with $\mathcal{O}(NL^2)$ precomputation.

Proteome-level PPI Screening. Whole-proteome screening poses a computational challenge due to the quadratic number of protein pairs. Recent pipelines address this using multi-stage filters coupled with structure prediction. In RF2-style workflows (Humphreys et al., 2024; Zhang et al., 2024), GPU-accelerated coevolutionary analysis (DCA (Ekeberg et al., 2013)) first prunes hundreds of millions of candidates; top pairs are then rescored with a lightweight RoseTTAFold2-based contact predictor, and the highest-confidence subset is finally evaluated with full AlphaFold2. These efforts primarily focus on accelerating structure prediction; for example, RF2-Lite (Humphreys et al., 2024) streamlines refinement with a two-track network that is $\sim 20\times$ faster than AlphaFold2, yet still requires explicit per-pair inference after pre-screening.

Our Position. RaftPPI is a residue-aware yet retrieval-friendly framework. It produces indexable protein embeddings whose inner products approximate residue-level interaction scoring, enabling ANN-based proteome screening without explicit per-pair inference. Compared with sequence-only models, RaftPPI attains stronger PPI classification while scaling to whole-proteome retrieval via precomputed embeddings. Compared with structure-prediction methods, it offers orders-of-magnitude faster screening by narrowing candidates first, after which shortlisted pairs can be rescored with accurate complex structure predictors.

3 Methodology

Figure 1 summarizes the RaftPPI framework. We first revisit a standard two-step pipeline for sequence-based PPI prediction (§3.1), then show how RaftPPI approximates residue-level modeling while *enabling scalable protein retrieval* by coupling a Gaussian-kernel interaction with low-rank (separable) attention and Structured Orthogonal Random Features (SORF) (§3.2). We optimize the model with an *adaptive negative reweighting* objective (§3.4) and enable proteome-scale search via vector indices over precomputed embeddings (§3.3).

Throughout, we consider a candidate protein pair (A, B) with residues indexed $1, \ldots, L_A$ and $1, \ldots, L_B$, respectively. Bold lowercase symbols (e.g., $\boldsymbol{z}$) denote vectors, bold uppercase symbols (e.g., $\boldsymbol{W}$) denote matrices, $\sigma(\cdot)$ is the logistic sigmoid, and $\|\cdot\|$ is the Euclidean norm.

3.1 A PIPELINE FOR TWO-STEP PPI PREDICTION

Since protein interactions occur at the residue level, a standard approach is a two-step pipeline (Sledzieski et al., 2021; Singh et al., 2022; Sledzieski et al., 2023) as illustrated in the upper panel of Figure 1: *Predict* residue-pair contact scores, then *aggregate* them into a protein-pair score. We refer to this two-step pipeline as *Pred&Agg* and define it as follows:

Residue-level contact matrix prediction. Given residue embeddings $\boldsymbol{z}_{A,i}, \boldsymbol{z}_{B,j} \in \mathbb{R}^d$ from a pretrained protein language model (PLM) (Lin et al., 2023), we compute *contact scores*

$$c_{i,j} = f(\boldsymbol{z}_{A,i}, \boldsymbol{z}_{B,j}) \in \mathbb{R}, \tag{1}$$

where $f(\cdot,\cdot)$ is typically an MLP or an inner product, producing a contact matrix $\boldsymbol{C} \in \mathbb{R}^{L_A \times L_B}$.

Protein-level interaction aggregation. This residue-level score matrix is pooled into a scalar logit:

$$\ell(A,B) = g(\boldsymbol{C}) \in \mathbb{R}, \tag{2}$$

where $g(\cdot)$ denotes a pooling operation (e.g., max/conv/2D attention).

There are many instances of this *Pred&Agg* pipeline. For example, D-SCRIPT (Sledzieski et al., 2021) uses Bepler & Berger embeddings (Bepler and Berger, 2019) with element-wise transforms to predict $\boldsymbol{C}$, then pools to a pair score. TT3D (Sledzieski et al., 2023) and Topsy-Turvy (Singh et al., 2022) incorporate graph-based supervision (Devkota et al., 2020) and 3Di structure encodings (van Kempen et al., 2024). While effective, these methods are *non-factorizable*: they rely on dense nonlinear residue–pair computations, yielding $\mathcal{O}(N^2L^2)$ complexity for N proteins of length L, which is prohibitive for proteome-scale screening. Next we show how RaftPPI approximates this pipeline as a dot product between *single-protein* embeddings.

3.2 KERNELIZED RESIDUE INTERACTIONS WITH LOW-RANK ATTENTION

As noted above, the nonlinearity in *Pred&Agg* yields strong accuracy but hinders retrieval because it requires explicit *pairwise* inputs. Our idea is to approximate these steps with *factorizable* ones where protein interaction scores are computed via dot-product between protein embeddings.

Predict (kernelized residue–residue scoring). We model residue–residue interactions with a Gaussian kernel

$$k_{\hat{\sigma}}\big(\boldsymbol{z}_{A,i}, \boldsymbol{z}_{B,j}\big) = \exp\left(-\frac{\|\boldsymbol{z}_{A,i} - \boldsymbol{z}_{B,j}\|^2}{2\hat{\sigma}^2}\right), \tag{3}$$

where $\hat{\sigma}^2$ is the kernel bandwidth that controls how quickly the kernel decays with residue-embedding distance. Smaller $\hat{\sigma}$ emphasizes very local residue matches, while larger values blend information across a wider neighborhood (see Appendix B.2 for the quantitative sweep). The kernel corresponds to an inner product in an infinite-dimensional Reproducing kernel Hilbert space (RKHS), enabling rich nonlinear scoring.

Aggregate (attention-weighted pooling). We aggregate the residue-level kernel scores into a protein-level logit via a weighted sum:

$$\ell(A,B) = \sum_{i=1}^{L_A} \sum_{j=1}^{L_B} s_{i,j} k_{\hat{\sigma}}\big(\boldsymbol{z}_{A,i}, \boldsymbol{z}_{B,j}\big), \tag{4}$$

where $s_{i,j}$ is the *attention weight*, determining the set of residue pairs of interest.

Factorizable low-rank attention. Low-rank (separable) attention is a standard strategy for reducing quadratic cost (e.g., Wang et al., 2020; Choromanski et al., 2021). We approximate the residue–pair attention with a rank-r separable form. Denote the residue embedding matrix as $\boldsymbol{Z}_A \in \mathbb{R}^{L_A \times d}$ and $\boldsymbol{Z}_B \in \mathbb{R}^{L_B \times d}$; for each $t \in \{1, \ldots, r\}$ a lightweight per-residue scorer $h_\theta^{(t)} : \mathbb{R}^d \to \mathbb{R}$ is applied *row-wise* to produce unnormalized importances, which we normalize with a softmax to obtain per-chain weights:

$$\boldsymbol{w}_A^{(t)} = \mathrm{softmax}\big(h_\theta^{(t)}(\boldsymbol{Z}_A)\big), \qquad \boldsymbol{w}_B^{(t)} = \mathrm{softmax}\big(h_\theta^{(t)}(\boldsymbol{Z}_B)\big). \tag{5}$$

Collecting columns gives $\boldsymbol{W}_A = [\,\boldsymbol{w}_A^{(1)} \cdots \boldsymbol{w}_A^{(r)}\,] \in \mathbb{R}_{\geq 0}^{L_A \times r}$ and $\boldsymbol{W}_B = [\,\boldsymbol{w}_B^{(1)} \cdots \boldsymbol{w}_B^{(r)}\,] \in \mathbb{R}_{\geq 0}^{L_B \times r}$, with nonnegative entries and each column summing to 1. We then set

$$s_{i,j} = \sum_{t=1}^{r} w_{A,i}^{(t)}\, w_{B,j}^{(t)} \quad \Longleftrightarrow \quad \boldsymbol{S} = \boldsymbol{W}_A \boldsymbol{W}_B^{\top}, \tag{6}$$

yielding a *factorizable* attention surface. In practice we instantiate $r{=}1$ (so $s_{i,j} = w_{A,i} w_{B,j}$), which achieves strong performance with minimal parameter cost.

3.3 Fast Inference with Random Fourier Features and Vector Search

Kernel approximation with Random Fourier Features. We approximate $k_{\hat{\sigma}}$ using Random Fourier Features (RFF) (Rahimi and Recht, 2007). Given d' target frequencies, let $\boldsymbol{W} \in \mathbb{R}^{d' \times d}$ and define

$$\psi(\boldsymbol{z}) = \tfrac{1}{\sqrt{d'}} \big[\cos(\boldsymbol{W}\boldsymbol{z}); \sin(\boldsymbol{W}\boldsymbol{z})\big] \in \mathbb{R}^{2d'}, \tag{7}$$

so that we have a factorized form to approximate the kernel as:

$$k_{\hat{\sigma}}(\boldsymbol{x}, \boldsymbol{y}) \approx \psi(\boldsymbol{x})^{\top} \psi(\boldsymbol{y}). \tag{8}$$

To construct $\boldsymbol{W}$ efficiently, we use Structured Orthogonal Random Features (SORF) (Yu et al., 2016). A SORF block of size $d \times d$ is

$$\tilde{\boldsymbol{W}} = \tfrac{\sqrt{d}}{\hat{\sigma}}\, \boldsymbol{H}\boldsymbol{D}_1\boldsymbol{H}\boldsymbol{D}_2\boldsymbol{H}\boldsymbol{D}_3, \tag{9}$$

where $\boldsymbol{H}$ is the normalized Walsh–Hadamard matrix and $\boldsymbol{D}_i$ are diagonal Rademacher sign-flip matrices. The rows of $\tilde{\boldsymbol{W}}$ satisfy $\mathbb{E}[\boldsymbol{w}\boldsymbol{w}^{\top}] = \hat{\sigma}^{-2}\boldsymbol{I}$, matching the Gaussian second moment and providing a low-variance RFF approximation of the Gaussian kernel. To obtain d' frequencies, we generate independent SORF blocks and concatenate their first d' rows to form $\boldsymbol{W} \in \mathbb{R}^{d' \times d}$.

Fast retrieval via factorizable scoring. Using Eq. 8 and the $r{=}1$ attention, we obtain

$$\begin{aligned} \ell(A, B) &= \sum_{i=1}^{L_A} \sum_{j=1}^{L_B} s_{i,j}\, k_{\hat{\sigma}}(\boldsymbol{z}_{A,i}, \boldsymbol{z}_{B,j}) \\ &\approx \sum_{i=1}^{L_A} \sum_{j=1}^{L_B} w_{A,i} w_{B,j}\, \psi(\boldsymbol{z}_{A,i})^{\top} \psi(\boldsymbol{z}_{B,j}) \\ &= \Big\langle \sum_{i=1}^{L_A} w_{A,i}\, \psi(\boldsymbol{z}_{A,i})\,, \sum_{j=1}^{L_B} w_{B,j}\, \psi(\boldsymbol{z}_{B,j}) \Big\rangle. \end{aligned} \tag{10}$$

Define the per-protein embeddings as

$$\hat{\boldsymbol{h}}_A = \sum_{i=1}^{L_A} w_{A,i}\, \psi(\boldsymbol{z}_{A,i}), \qquad \hat{\boldsymbol{h}}_B = \sum_{j=1}^{L_B} w_{B,j}\, \psi(\boldsymbol{z}_{B,j}), \tag{11}$$

which yields the factorizable approximation that computes the logit $\ell(A, B)$ in a dot-product form:

$$\ell(A, B) := \langle \hat{\boldsymbol{h}}_A, \hat{\boldsymbol{h}}_B \rangle. \tag{12}$$

We fix the same SORF transform $\boldsymbol{W}$ at training and inference for alignment, and store $\hat{\boldsymbol{h}}$ for approximate k-nearest neighbor search with HNSW (Malkov and Yashunin, 2020) (inner-product retrieval in the transformed space).

3.4 Training Objective

As noted in the Introduction, experimentally verified non-interactions are rare, so negatives are often *constructed* and vary widely in informativeness (Neumann et al., 2022; Zhao et al., 2022). Heuristic constructions (e.g., enforcing different cellular compartments) often produce overly easy, biased negatives, leading to overly optimistic results. Meanwhile, co-localized, functionally related, or topology-aware choices tend to be harder and more informative (Ben-Hur and Noble, 2006; Park and Marcotte, 2011; 2012; Zhang et al., 2018). Motivated by this, we adopt *adaptive negative weighting*, where the relative contribution of each negative is automatically determined by the model's own confidence, allowing harder negatives to exert greater influence.

Let $\ell(A, B) = \langle \hat{\boldsymbol{h}}_A, \hat{\boldsymbol{h}}_B \rangle$ denote the logit for a pair (A, B). Over a minibatch, let $\mathcal{P}$ and $\mathcal{N}$ be the index sets of positive and negative pairs, respectively. Inspired by self-adversarial training in knowledge-graph reasoning (Sun et al., 2019), we define temperature-scaled weights over negatives

$$p_i = \frac{\exp\big(\tau\, \ell_i\big)}{\sum\limits_{j \in \mathcal{N}} \exp\big(\tau\, \ell_j\big)}, \qquad i \in \mathcal{N}, \quad \tau \geq 0, \tag{13}$$

and stop gradients through p_i in practice. Intuitively, p_i reflects the model's (normalized) confidence that a negative pair is actually positive, i.e. higher p_i thus identifies *harder* negatives. We then combine a standard positive term with an adaptively reweighted negative term:

$$\mathcal{L} = \frac{1}{2}\left[-\frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \log \sigma(\ell_p) - \sum_{i \in \mathcal{N}} p_i \, \log \sigma(-\ell_i) \right]. \tag{14}$$

When $\tau = 0$, Eq. 14 reduces to balanced BCE; as $\tau \to \infty$, it focuses on the hardest negative. In practice, $\tau = 4$ offers a good trade-off, as shown in Appendix B.4.

3.5 Computational Complexity

Consider a proteome with N proteins of average length L. PLM embedding dominates at $\mathcal{O}(NL^2)$. Our mapping/aggregation adds $\mathcal{O}\big(L\,(d \log d + d')\big)$ per protein (vs. $\mathcal{O}(Ldd')$ for dense RFF), which is linear in L and minor in practice. After caching $\hat{\boldsymbol{h}}$, HNSW indexing is $\mathcal{O}(N \log N)$, queries grow polylogarithmically in N, and memory is $\mathcal{O}(N)$. Hence the end-to-end complexity is $\mathcal{O}(NL^2)$, with indexing/search negligible; empirically, top-20% retrieval completes in $\sim$6 minutes for $N \approx 10^4$ (see §4.3, Table 3).

4 Experiments

4.1 Experimental Setup

Datasets. As discussed in (Bernett et al., 2024), naïve PPI splits and datasets are prone to *data leakage* and confounding from sequence similarity and node-degree biases, which can yield over-optimistic performance and poor transfer. We therefore adopt the 7 processed datasets/splits in (Bernett et al., 2024) that (i) remove near-duplicate or homologous sequences across train/validation and test sets, minimizing the impact of raw sequence similarity, and (ii) control protein occurrence frequency so that hub proteins do not trivially inflate accuracy via degree priors. This setting makes PPI prediction more realistic and challenging. The datasets span two species: yeast (Guo, Du) and human (Huang, D-SCRIPT, Pan, Richoux, Gold). Their statistics are shown in Table 1. In Appendix B.6 we further evaluate RaftPPI and baselines on the larger-scale PiNUI-human and PiNUI-yeast datasets (Dubourg-Felonneau et al., 2023).

Baselines. We evaluate 10 PPI classifiers spanning classical sequence models (D-SCRIPT (Sledzieski et al., 2021), DeepFE (Yao et al., 2019), Richoux-FC/LSTM (Richoux et al., 2019), Topsy–Turvy (Singh et al., 2022), SPRINT (Li and Ilie, 2017)) and PLM-based methods (ESM2-MLP (Sledzieski et al., 2024), TUnA (Ko et al., 2024), PLM-Interact (Liu et al., 2024)). For all PLM baselines and for RaftPPI, we use ESM2-8M as the backbone; we also include an unsupervised ESM2-NoFT baseline (dot product over `[CLS]` embeddings). Prior work (Fournier et al., 2024)

Table 1: Dataset statistics of seven PPI datasets spanning two species (human and yeast).

Dataset	Species	Train			Val			Test			Total
		Pos	Neg	Total	Pos	Neg	Total	Pos	Neg	Total	
GUO	Yeast	2,088	2,088	4,176	232	232	464	861	861	1,722	6,362
DU	Yeast	6,536	6,486	13,022	698	748	1,446	2,421	2,421	4,842	19,310
HUANG	Human	1,094	1,075	2,169	111	130	241	713	713	1,426	3,836
D-SCRIPT	Human	12,218	122,165	134,383	1,356	13,575	14,931	8,467	84,670	93,137	242,451
PAN	Human	14,069	14,022	28,091	1,537	1,584	3,121	4,575	4,575	9,150	40,362
RICHOUX	Human	17,873	17,798	35,671	1,944	2,019	3,963	5,167	5,167	10,334	49,968
GOLD	Human	81,596	81,596	163,192	29,630	29,630	59,260	26,024	26,024	52,048	274,500

and our analysis at Appendix B.1 show that scaling ESM2 (35M/150M/650M) does not consistently improve these tasks; moreover, proteome-scale retrieval with per-pair PLM inference is already expensive at the 8M scale, consuming GPU-months computation time (see Table 3). In this case, we use ESM2-8M for the best performance/throughput trade-off in our evaluations.

4.2 PROTEIN-PROTEIN INTERACTION CLASSIFICATION

Table 2: Test AUROC Performance (%) of competing methods on the seven PPI datasets. Higher is better; the rightmost column shows the mean across collections.

Method	D-SCRIPT	Huang	Pan	Richoux	Gold	Guo	Du	Average
D-SCRIPT (Sledzieski et al., 2021)	81.99	65.72	68.44	59.15	49.91	47.14	50.92	60.47
DeepFE (Yao et al., 2019)	61.66	56.93	53.70	57.43	53.21	58.21	56.05	56.74
Richoux-FC (Richoux et al., 2019)	47.53	59.06	52.39	59.72	53.53	55.81	59.89	55.42
Richoux-LSTM (Richoux et al., 2019)	50.67	56.56	47.81	50.37	49.16	51.55	56.37	51.78
SPRINT (Li and Ilie, 2017)	64.62	46.71	43.39	55.71	51.50	48.80	51.80	51.79
Topsy-Turvy (Singh et al., 2022)	75.38	55.52	65.80	48.67	58.74	43.65	61.60	58.48
ESM2-NoFT (Lin et al., 2023)	75.01	58.63	59.51	63.26	57.85	62.87	57.36	62.07
ESM2-MLP (Sledzieski et al., 2024)	82.83	**73.34**	72.89	77.48	56.35	83.54	73.34	74.25
TUnA (Ko et al., 2024)	83.38	66.66	**77.06**	76.73	52.55	69.81	69.37	70.79
PLM-Interact (Liu et al., 2024)	**84.77**	69.69	73.03	**78.66**	65.00	79.60	**75.20**	75.14
RaftPPI	82.06	72.20	74.21	69.88	**68.69**	**84.93**	75.06	**75.29**

The PPI classification results across the seven datasets are reported in Table 2. We observe that methods without pretrained PLMs—D-SCRIPT, DeepFE, Richoux-FC/LSTM, Topsy-Turvy, and SPRINT—perform substantially worse than ESM2-based models, all falling below even the unsupervised ESM2-NoFT baseline. This is because they rely largely on sequence similarity and node-degree information, which fails under controlled splits (Bernett et al., 2024), whereas ESM-based models benefit from large-scale pretraining, where structural properties such as secondary structure can be inferred from embeddings (Rives et al., 2021).

For ESM2-based baselines, all models finetuned for PPI outperform ESM2-NoFT. PLM-Interact achieves the strongest results, likely due to its early-fusion design, which allows deeper layers to jointly model cross-protein interactions. In contrast, TUnA and ESM2-MLP fuse only at intermediate or final layers, limiting their ability to capture joint interactions. This echoes the early-fusion advantage reported in other domains (Snoek et al., 2005). Meanwhile, RaftPPI attains the best average performance, attributable to residue-level interaction modeling and adaptive negative weighting, which we further discuss in §4.4.

4.3 PROTEOME INTERACTION RETRIEVAL

Compared with binary PPI classification, PPI retrieval more closely reflects real-world applications: interactions in proteomes are *sparse* and highly *imbalanced* (negatives dominate), and one must screen an entire candidate proteome to identify true interactors for a query protein. For computational tractability, we sample 100 query proteins per dataset and use the models trained in Section 4.2 to retrieve positives on the test split. We compare RaftPPI to PLM (ESM2) baselines (ESM2-NoFT, TUnA, PLM-Interact, ESM2-MLP) and to RaftPPI-P, a special version that removes the residue-level design of RaftPPI, which predicts PPI using the dot-product of `[CLS]` token embeddings.

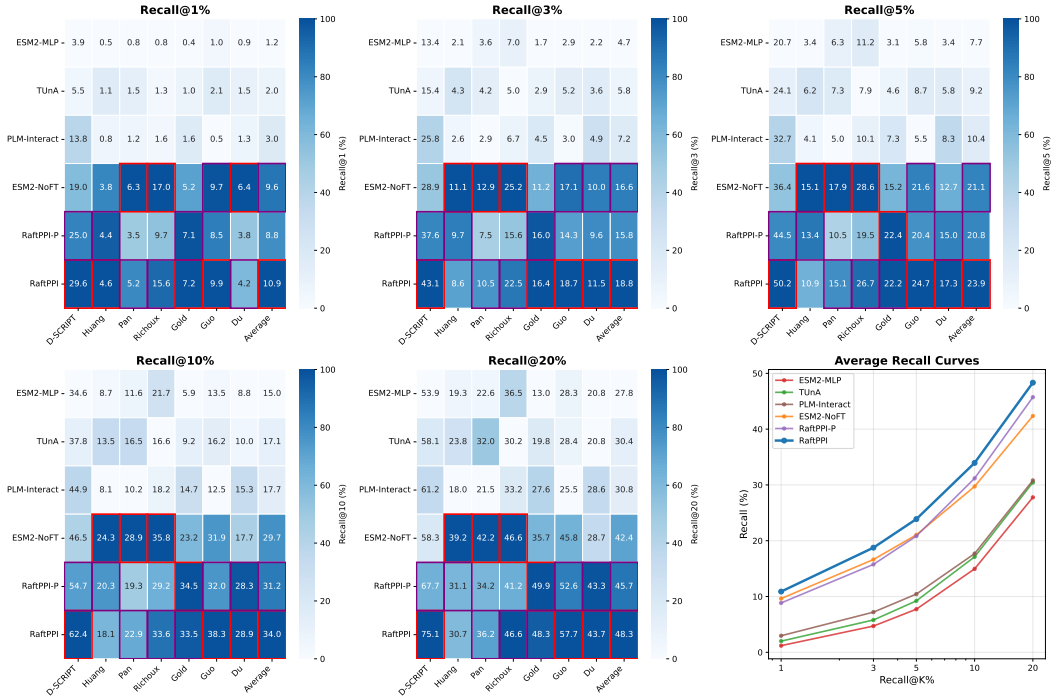

Figure 2: **Proteome retrieval with residue-level fidelity and scalable retrieval.** Heatmaps report Recall@$K\%$ for $K \in \{1, 3, 5, 10, 20\}$ across methods and datasets, with per-dataset *normalized* color scales (best=100%, worst=0%). Red rectangles mark the best method; purple rectangles mark the second best. The sixth panel shows average recall curves across datasets.

Type	Model	Encoding (s)	Recall@1/3/5/10/20% (s)	AUROC (%)	Recall@20% est. / full (%)
Unfactorizable	ESM2-MLP	NA	1766576	74.25	27.77 / NA
	TUnA	NA	3833646	70.79	30.44 / NA
	PLM-Interact	NA	12827660	75.14	30.81 / NA
Factorizable	ESM2-NoFT	105	54 / 74 / 99 / 170 / 259	62.07	42.37 / 41.72
	RaftPPI-P	54	40 / 48 / 75 / 118 / 187	71.90	45.73 / 43.83
	RaftPPI	102	49 / 70 / 98 / 157 / 241	**75.29**	**48.33 / 47.91**

Table 3: Human proteome retrieval efficiency (Recall@K% end-to-end time) and average classification/retrieval performance. Factorizable methods that reuse single-protein embeddings can build an index once and then retrieve via HNSW (Encoding time shows the one-time cost). Due to intractable computing time for unfactorizable methods, we estimate the recall performance (denoted as est.) and inference time using 100 query proteins. Times are estimated total seconds for the full set of queries on proteome; measured on an A100 GPU.

Retrieval performance. As shown in Figure 2, non-factorizable methods—i.e., models that jointly encode protein pairs such as ESM2-MLP, TUnA, and PLM-Interact—achieve strong binary PPI classification but do not outperform the simple ESM2-NoFT baseline in retrieval. We hypothesize that this gap arises because the data splits are explicitly designed to minimize sequence similarity (Bernett et al., 2024), limiting the models' ability to exploit correlations between sequence similarity and structural interaction. In contrast, factorizable approaches naturally capture sequence similarity through embedding dot products. Among these, RaftPPI consistently ranks first or second across datasets and recall thresholds, and its improvements over RaftPPI-P highlight the benefit of explicitly modeling residue-level interactions.

Retrieval efficiency. Table 3 reports runtime comparisons. Non-factorizable methods—ESM2-MLP, TUnA, and PLM-Interact—are prohibitively slow because they require per-pair inference. The strongest of these, PLM-Interact, performs well on both classification and retrieval, yet demands 148.47 A100 GPU-days (≈4.9 months) to screen the human proteome, underscoring the impracticality

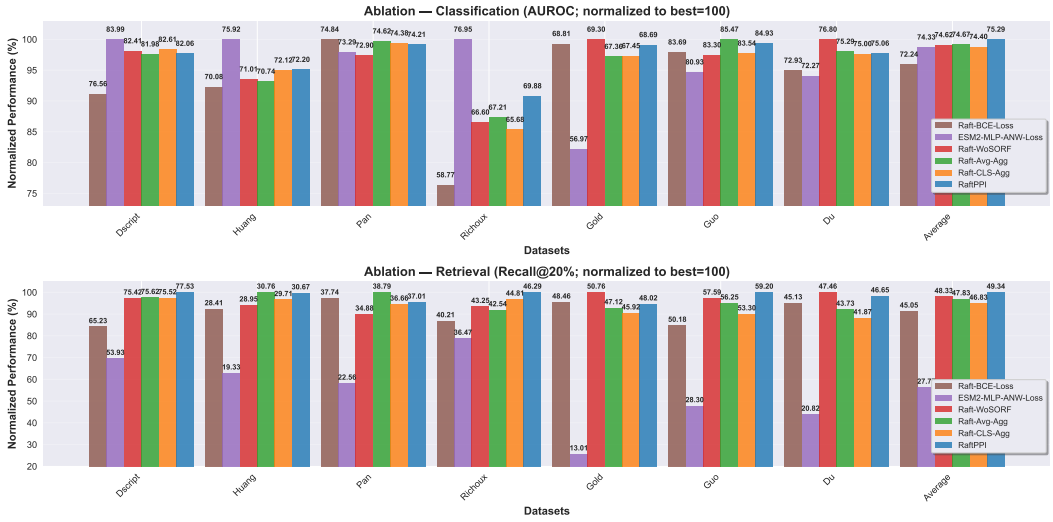

Figure 3: **Ablation Study.** Scores are normalized to the best method per dataset (100%=peak). *Top:* classification AUROC. *Bottom:* retrieval Recall@20%. The full model (RaftPPI) attains the best or second–best performance in every case; removing any single design choice causes a clear drop.

of exhaustive search. By contrast, RaftPPI compresses each protein once and performs approximate nearest-neighbor search, reducing proteome retrieval from *GPU months* to *minutes*. Overall, RaftPPI delivers the best balance of retrieval accuracy, classification performance, and efficiency.

4.4 Ablation Study

We focus on four research questions and validate our core designs. **RQ1 (residue-level modeling):** Does explicit residue-level modeling help? **RQ2 (kernel):** Does replacing a linear dot product with a Gaussian kernel (approximated via SORF) improve performance? **RQ3 (aggregation):** How should residue scores be aggregated most effectively? **RQ4 (adaptive negative weighting):** Is adaptive negative weighting effective in helping the model assign greater weight to harder negatives?

We perform ablation studies on these research questions and show the results in Figure 3. **RQ1 (residue-level modeling):** The coarse ESM2-MLP baseline encodes proteins and predicts interactions from concatenated embeddings. Even when augmented with our adaptive loss (ESM2-MLP-ANW-Loss), it competes on small classification splits but collapses on retrieval (e.g., -46% Recall@20% on Gold), confirming the necessity of residue-level reasoning for large-candidate screening. **RQ2 (kernel):** Replacing the Gaussian kernel with a linear dot product (Raft-WoSORF) consistently reduces AUROC and Recall@20%, showing the benefit of kernelized interactions. We additionally discuss the impact of Gaussian bandwidth $\hat{\sigma}$ in Appendix B.2. **RQ3 (aggregation):** Averaging (Raft-Avg-Agg) or using a `[CLS]` score (Raft-CLS-Agg) are slightly weaker than attention, indicating that attention helps identify PPIs. **RQ4 (adaptive negative weighting):** Switching to uniform BCE (Raft-BCE) degrades AUROC and Recall@20%, especially on the D-SCRIPT dataset where negatives outnumber positives by roughly $10\times$, highlighting the value of prioritizing hard negatives (we provide a detailed ablation study of the temperature τ in Appendix B.4). Collectively, these ablations validate each design choice in RaftPPI, demonstrating the effectiveness of kernelized residue interactions, SORF-based kernels, attention-based aggregation, and adaptive negative weighting.

5 Conclusion

Conclusion. We introduced RaftPPI, a residue-level framework for scalable proteome-wide PPI retrieval. By combining a kernelized interaction module with low-rank attention aggregation, RaftPPI approximates residue-level interactions in a factorizable form, producing compact indexable protein embeddings that support efficient retrieval. In addition, we incorporate adaptive negative weighting, which prioritizes harder negatives during training and further strengthens model performance. RaftPPI

achieves state-of-the-art performance on both PPI classification and retrieval, while enabling residue-aware and retrieval-friendly screening at proteome scale.

Limitation. Although RaftPPI achieves strong retrieval efficiency, the kernel approximation and rank-r attention introduce inductive biases that simplify residue–residue interactions into a low-rank form, which may under-represent subtle allosteric effects or conformational rearrangements at complex interfaces. Moreover, RaftPPI is trained as a sequence-based PPI classifier on pairwise labels without complex-level structural supervision, so it does not directly observe ground truth residue contact patterns and may inherit dataset biases in assay type, species coverage, and interaction density.

Future work. An important next step is to develop *structure-aware pretraining* that incorporates complex-level geometric signals (e.g., residue contact maps, interface distances, or docking poses), allowing the kernel and attention to learn physically grounded interaction patterns while preserving a factorizable retrieval head. Beyond supervision, integrating *structure-backed retrieval*, where coarse vector search proposes candidates that are subsequently refined or rescored by structure prediction models such as AlphaFold3, could couple RaftPPI-style screening with more accurate yet expensive structural modeling. Finally, extending our framework to capture condition- or state-specific PPIs (e.g., tissue, perturbation, or disease context) and to operate over even larger cross-species proteomes with dynamic, updatable indices are promising directions toward truly comprehensive, context-aware proteome-wide PPI retrieval.

REFERENCES

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N. Kinch, R. David Schaeffer, Claudia Millán, Hahnbeom Park, Carson Adams, Caleb R. Glassman, Andy DeGiovanni, Jose Henrique Pereira, Andrew V. Rodrigues, Albertha A. van Dijk, Anna C. Ebrecht, Daren J. Opperman, Thomas Sagmeister, Christoph Buhlheller, Tea Pavkov-Keller, Manoj K Rathinaswamy, Prajwal Dalwadi, Colin K. Yip, John E. Burke, K. Christopher Garcia, Nick V. Grishin, Paul D. Adams, Randy J. Read, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021. doi: 10.1126/science.abj8754.

Asa Ben-Hur and William Stafford Noble. Choosing negative examples for the prediction of protein-protein interactions. *BMC Bioinform.*, 7(S-1), 2006. doi: 10.1186/1471-2105-7-S1-S2. URL https://doi.org/10.1186/1471-2105-7-S1-S2.

Tristan Bepler and Bonnie Berger. Learning protein sequence embeddings using information from structure. In *Proceedings of the International Conference on Learning Representations (ICLR) 2019*, 2019.

Judith Bernett, David B Blumenthal, and Markus List. Cracking the black box of deep sequence-based protein–protein interaction prediction. *Briefings in Bioinformatics*, 25(2):bbae076, 03 2024. ISSN 1477-4054. doi: 10.1093/bib/bbae076. URL https://doi.org/10.1093/bib/bbae076.

Muhao Chen, Chelsea J.-T. Ju, Guangyu Zhou, Xuelu Chen, Tianran Zhang, Kai-Wei Chang, Carlo Zaniolo, and Wei Wang. Multifaceted protein–protein interaction prediction based on Siamese residual RCNN. *Bioinformatics*, 35(14):i305–i314, July 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz328.

Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. In *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=Ua6zuk0WRH.

Kapil Devkota, James M. Murphy, and Lenore J. Cowen. GLIDE: Combining local methods and diffusion state embeddings to predict missing interactions in biological networks. *Bioinformatics*, 36:i464–i473, July 2020. ISSN 1367-4803. doi: 10.1093/bioinformatics/btaa459.

Geoffroy Dubourg-Felonneau, Daniel Mitiku Wesego, Eyal Akiva, and Ranjani Varadan. PiNUI: A dataset of protein–protein interactions for machine learning. In *NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development*, 2023. doi: 10.1101/2023.12.12.571298. bioRxiv preprint 10.1101/2023.12.12.571298.

Magnus Ekeberg, Cecilia Lövkvist, Yueheng Lan, Martin Weigt, and Erik Aurell. Improved contact prediction in proteins: Using pseudolikelihoods to infer Potts models. *Physical Review E*, 87(1): 012707, January 2013. ISSN 1539-3755, 1550-2376. doi: 10.1103/PhysRevE.87.012707.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rihawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Bernardino Romera-Paredes, Pushmeet Kohli, John Jumper, Demis Hassabis, Augustin Žídek, Russell Bates, Sam Blackwell, Jason Yim, Olaf Ronneberger, Sebastian Bodenstein, Michal Zielinski, Alex Bridgland, Anna Potapenko, Andrew Cowie, Kathryn Tunyasuvunakool, Rishub Jain, and Ellen Clancy. Protein complex prediction with alphafold-multimer. *bioRxiv*, page 2021.10.04.463034, 2021. doi: 10.1101/2021.10.04.463034.

Quentin Fournier, Robert M. Vernon, Almer van der Sloot, Benjamin Schulz, Sarath Chandar, and Christopher James Langmead. Protein Language Models: Is Scaling Necessary?, September 2024. URL https://www.biorxiv.org/content/10.1101/2024.09.23.614603v1.

Ziqi Gao, Chenran Jiang, Jiawen Zhang, Xiaosen Jiang, Lanqing Li, Peilin Zhao, Huanming Yang, Yong Huang, and Jia Li. Hierarchical graph learning for protein–protein interaction. *Nature Communications*, 14(1):1093, February 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-36736-1. URL https://doi.org/10.1038/s41467-023-36736-1.

Yuxin Huang, Stefan Wuchty, Yuedong Zhou, and Zemin Zhang. Sgppi: structure-aware prediction of protein–protein interactions in rigorous conditions with graph convolutional network. *Briefings in Bioinformatics*, 24:bbad020, 2023. doi: 10.1093/bib/bbad020.

Ian R. Humphreys, Jing Zhang, Minkyung Baek, Yaxi Wang, Aditya Krishnakumar, Jimin Pei, Ivan Anishchenko, Catherine A. Tower, Blake A. Jackson, Thulasi Warrier, Deborah T. Hung, S. Brook Peterson, Joseph D. Mougous, Qian Cong, and David Baker. Protein interactions in human pathogens revealed through deep learning. *Nature Microbiology*, 9(10):2642–2652, September 2024. ISSN 2058-5276. doi: 10.1038/s41564-024-01791-x. URL https://www.nature.com/articles/s41564-024-01791-x.

Kanchan Jha, Sriparna Saha, and Hiteshi Singh. Prediction of protein–protein interaction using graph neural networks. *Scientific Reports*, 12(1):8360, May 2022. ISSN 2045-2322. doi: 10.1038/s41598-022-12201-9. URL https://www.nature.com/articles/s41598-022-12201-9. Publisher: Nature Publishing Group.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russell Bates, Augustin Zidek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Marta Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596 (7873):583–589, 2021. doi: 10.1038/s41586-021-03819-2.

John Jumper, Richard Evans, Alexander Pritzel, Demis Hassabis, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 2024. doi: 10.1038/s41586-024-07487-w.

Young Su Ko, Jonathan Parkinson, Cong Liu, and Wei Wang. TUnA: an uncertainty-aware transformer model for sequence-based protein–protein interaction prediction. *Briefings in Bioinformatics*, 25 (5):bbae359, September 2024. ISSN 1477-4054. doi: 10.1093/bib/bbae359. URL https://doi.org/10.1093/bib/bbae359.

Yiwei Li and Lucian Ilie. SPRINT: Ultrafast protein-protein interaction prediction of the entire human interactome, May 2017. URL http://arxiv.org/abs/1705.06848. arXiv:1705.06848 [q-bio].

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023. doi: 10.1126/science.ade2574.

Dan Liu, Francesca Young, Kieran D. Lamb, Adalberto Claudio Quiros, Alexandrina Pancheva, Crispin Miller, Craig Macdonald, David L. Robertson, and Ke Yuan. PLM-interact: Extending protein language models to predict protein-protein interactions, November 2024.

Joseph Loscalzo. Molecular interaction networks and drug development: Novel approach to drug target identification and drug repositioning. *The FASEB Journal*, 37(1):e22660, 2023. doi: 10.1096/fj.202201683R. URL https://doi.org/10.1096/fj.202201683R.

Guofeng Lv, Zhiqiang Hu, Yanguang Bi, and Shaoting Zhang. Learning unknown from correlations: Graph neural network for inter-novel-protein interaction prediction. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2021. URL https://arxiv.org/abs/2105.06709. See also arXiv:2105.06709.

Ali Madani, Bryan McCann, Nikhil Naik, Nitish Shirish Keskar, Namrata Anand, Raphael R. Eguchi, Po-Ssu Huang, and Richard Socher. ProGen: Language Modeling for Protein Generation, March 2020.

Yu. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020. doi: 10.1109/TPAMI.2018.2889473.

Elahe Nasiri, Kamal Berahmand, Mehrdad Rostami, and Mohammad Dabiri. A novel link prediction algorithm for protein-protein interaction networks by attributed graph embedding. *Comput. Biol. Medicine*, 137:104772, 2021. doi: 10.1016/J.COMPBIOMED.2021.104772. URL https://doi.org/10.1016/j.compbiomed.2021.104772.

Don Neumann, Soumyadip Roy, Fayyaz ul Amir Afsar Minhas, and Asa Ben-Hur. On the choice of negative examples for prediction of host-pathogen protein interactions. *Frontiers Bioinform.*, 2, 2022. doi: 10.3389/FBINF.2022.1083292. URL https://doi.org/10.3389/fbinf.2022.1083292.

Erik Nijkamp, Jeffrey Ruffolo, Eli N. Weinstein, Nikhil Naik, and Ali Madani. ProGen2: Exploring the Boundaries of Protein Language Models, June 2022.

Jeffrey Ouyang-Zhang, Chengyue Gong, Yue Zhao, Philipp Krähenbühl, Adam R. Klivans, and Daniel J. Diaz. Distilling structural representations into protein sequence models. In *Proceedings of the International Conference on Learning Representations (ICLR) 2025*, 2025.

Yungki Park and Edward M. Marcotte. Revisiting the negative example sampling problem for predicting protein–protein interactions. *Bioinformatics*, 27(21):3024–3028, September 2011. ISSN 1367-4803. doi: 10.1093/bioinformatics/btr514. URL https://doi.org/10.1093/bioinformatics/btr514. tex.eprint: https://academic.oup.com/bioinformatics/article-pdf/27/21/3024/48862842/bioinformatics_27_21_3024.pdf.

Yungki Park and Edward M Marcotte. Flaws in evaluation schemes for pair-input computational predictions. *Nature Methods*, 9(12):1134–1136, December 2012. ISSN 1548-7105. doi: 10.1038/nmeth.2259. URL https://doi.org/10.1038/nmeth.2259.

Ali Rahimi and Benjamin Recht. Random Features for Large-Scale Kernel Machines. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007. URL https://papers.nips.cc/paper_files/paper/2007/hash/013a006f03dbc5392effeb8f18fda755-Abstract.html.

Florian Richoux, Charlène Servantie, Cynthia Borès, and Stéphane Téletchéa. Comparing two deep learning sequence-based models for protein-protein interaction prediction, January 2019. URL http://arxiv.org/abs/1901.06268. arXiv:1901.06268 [cs].

Anna Ritz, Christopher L. Poirel, Allison N. Tegge, Nicholas Sharp, Kelsey Simmons, Allison Powell, Shiv D. Kale, and T. M. Murali. Pathways on demand: automated reconstruction of human signaling networks. *npj Systems Biology and Applications*, 2:16002, 2016. doi: 10.1038/npjsba.2016.2. URL https://www.nature.com/articles/npjsba20162.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, April 2021. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.2016239118. URL https://pnas.org/doi/full/10.1073/pnas.2016239118.

Roded Sharan, Igor Ulitsky, and Ron Shamir. Network-based prediction of protein function. *Molecular Systems Biology*, 3:88, 2007. doi: 10.1038/msb4100129. URL https://pubmed.ncbi.nlm.nih.gov/17353930/.

Rohit Singh, Kapil Devkota, Samuel Sledzieski, Bonnie Berger, and Lenore Cowen. Topsy-Turvy: Integrating a global view into sequence-based PPI prediction. *Bioinformatics*, 38:i264–i272, June 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btac258.

Samuel Sledzieski, Rohit Singh, Lenore Cowen, and Bonnie Berger. Sequence-based prediction of protein-protein interactions: a structure-aware interpretable deep learning model. *bioRxiv*, 2021. doi: 10.1101/2021.01.22.427866. URL https://www.biorxiv.org/content/early/2021/01/25/2021.01.22.427866.

Samuel Sledzieski, Kapil Devkota, Rohit Singh, Lenore Cowen, and Bonnie Berger. TT3D: Leveraging precomputed protein 3D sequence models to predict protein–protein interactions. *Bioinformatics*, 39(11):btad663, November 2023. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad663.

Samuel Sledzieski, Meghana Kshirsagar, Minkyung Baek, Rahul Dodhia, Juan Lavista Ferres, and Bonnie Berger. Democratizing protein language models with parameter-efficient fine-tuning. *Proceedings of the National Academy of Sciences*, 121(26):e2405840121, June 2024. doi: 10.1073/pnas.2405840121. URL https://www.pnas.org/doi/10.1073/pnas.2405840121. Publisher: Proceedings of the National Academy of Sciences.

Cees Snoek, Marcel Worring, and Arnold W. M. Smeulders. Early versus late fusion in semantic video analysis. In HongJiang Zhang, Tat-Seng Chua, Ralf Steinmetz, Mohan S. Kankanhalli, and Lynn Wilcox, editors, *Proceedings of the 13th ACM International Conference on Multimedia, Singapore, November 6-11, 2005*, pages 399–402. ACM, 2005. doi: 10.1145/1101149.1101236. URL https://doi.org/10.1145/1101149.1101236.

Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. SaProt: Protein Language Modeling with Structure-aware Vocabulary, October 2023.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *Proceedings of the International Conference on Learning Representations (ICLR) 2019*, 2019.

Michel van Kempen, Stephanie S. Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron L. M. Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with Foldseek. *Nature Biotechnology*, 42(2):243–246, February 2024. ISSN 1546-1696. doi: 10.1038/s41587-023-01773-0.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-Attention with Linear Complexity, June 2020. URL http://arxiv.org/abs/2006.04768. arXiv:2006.04768 [cs].

Sen Yang, Peng Cheng, Yang Liu, Dawei Feng, and Shengqi Wang. Exploring the Knowledge of an Outstanding Protein to Protein Interaction Transformer. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 21(5):1287–1298, March 2024. ISSN 1545-5963. doi: 10.1109/TCBB.2024.3381825. URL https://doi.org/10.1109/TCBB.2024.3381825.

Yu Yao, Xiuquan Du, Yanyu Diao, and Huaixu Zhu. An integration of deep learning with feature embedding for protein-protein interaction prediction. *PeerJ*, 7:e7126, 2019. ISSN 2167-8359. doi: 10.7717/peerj.7126.

Felix X. Yu, Ananda Theertha Suresh, Krzysztof Marcin Choromanski, Daniel N. Holtmann-Rice, and Sanjiv Kumar. Orthogonal random features. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *NeurIPS*, pages 1975–1983, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/53adaf494dc89ef7196d73636eb2451b-Abstract.html.

Jing Zhang, Ian R. Humphreys, Jimin Pei, Jinuk Kim, Chulwon Choi, Rongqing Yuan, Jesse Durham, Siqi Liu, Hee-Jung Choi, Minkyung Baek, David Baker, and Qian Cong. Computing the human interactome. *bioRxiv*, 2024. doi: 10.1101/2024.10.01.615885. URL https://www.biorxiv.org/content/early/2024/10/01/2024.10.01.615885.

Long Zhang, Guoxian Yu, Maozu Guo, and Jun Wang. Predicting protein-protein interactions using high-quality non-interacting pairs. *BMC Bioinformatics*, 19(19):525, December 2018. ISSN 1471-2105. doi: 10.1186/s12859-018-2525-3. URL https://doi.org/10.1186/s12859-018-2525-3.

Nan Zhao, Maji Zhuo, Kun Tian, and Xinqi Gong. Protein–protein interaction and non-interaction predictions using gene sequence natural vector. *Communications Biology*, 5(1):652, July 2022. ISSN 2399-3642. doi: 10.1038/s42003-022-03617-0. URL https://doi.org/10.1038/s42003-022-03617-0.

A Implementation Details

All experiments use five random seeds ({0, 1, 2, 3, 4}); unless noted, we report the mean (and standard deviation when available) across seeds. Results for D-SCRIPT (Sledzieski et al., 2021), DeepFE (Yao et al., 2019), Richoux-FC/LSTM (Richoux et al., 2019), and Topsy-Turvy (Singh et al., 2022) are taken from the benchmarking study of Bernett et al. (2024) ; following their guidance, we apply minimal tuning to baselines based on validation performance. For our method, we use a single configuration across datasets: AdamW with learning rate 1e-4, 2048 random Fourier features, Gaussian kernel bandwidth $\hat{\sigma} = 0.5$ (selected via Appendix B.2), and adversarial temperature $\tau = 4$ (see Appendix B.4). Software environment: Python 3.10; PyTorch 2.5 with CUDA 11.8. Anonymized code and data to reproduce the results are included in the supplementary materials.

B Additional Experiments

B.1 Model-Scale Selection

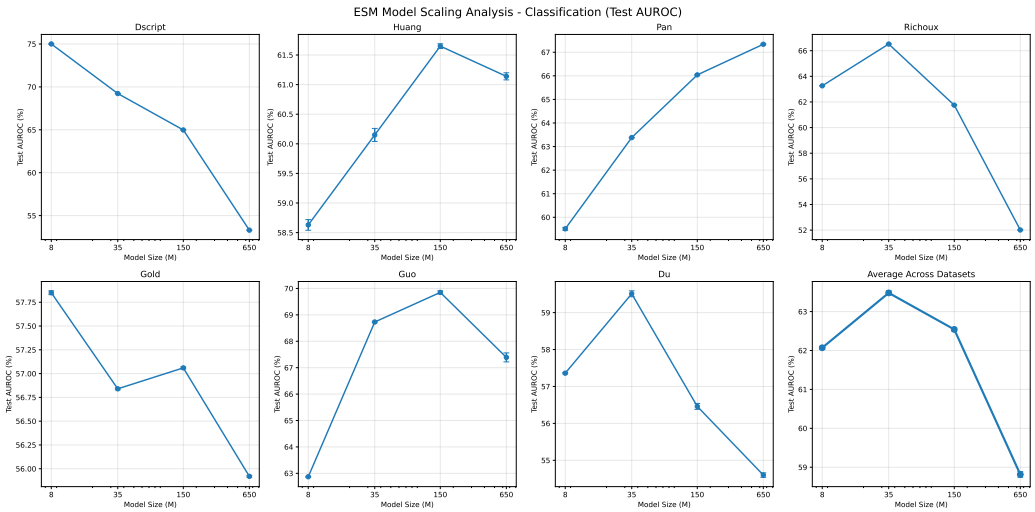

Figure 4: ESM2 model–scaling analysis for PPI binary-classification (Test AUROC) across model sizes and datasets. Increasing the parameter count beyond 8M does not consistently improve performance.

Following the observation of Fournier et al. (2024) that larger protein LMs do not necessarily yield better results, we conduct a systematic scaling study on ESM2 (Lin et al., 2023). We evaluate four checkpoints: 8M, 35M, 150M, and 650M parameters across seven PPI datasets spanning human and yeast. Protein pairs are scored by the dot product of their `[CLS]` embeddings, providing an unsupervised measure of scaling performance. As shown in Figures 4 and 5, both Test AUROC and Recall@K% change little with model size on nearly all datasets.

Given that inference time for pairwise-encoding models rises steeply with both model size and quadratic pairwise scoring (e.g., PLM-Interact (Liu et al., 2024) requires *GPU-months* to search the human proteome even with an 8M model due to its pairwise encoding; see Table 3), we adopt the 8M ESM2 checkpoint as the backbone for all PLM-based baselines and for RaftPPI, balancing predictive performance and efficiency.

B.2 Ablation on Gaussian Kernel Bandwidth

The Gaussian kernel in Eq. 3 controls how strictly residue-level interactions are determined: smaller $\hat{\sigma}$ values confine each residue to interact only with very close neighbors (capturing sharp, local interfaces), whereas larger $\hat{\sigma}$ allows larger interaction scores over a broader neighborhood. Figure 6 sweeps $\hat{\sigma} \in \{0.125, 0.25, 0.5, 1, 2, 4, 8\}$ and averages the metrics within human and yeast datasets. From the results of both PPI classification and retrieval tasks, both very small and very large

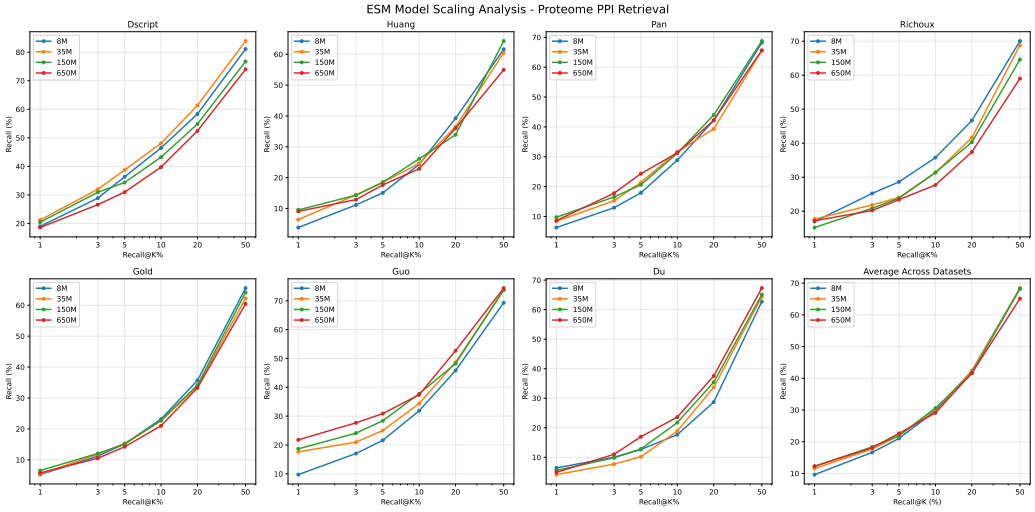

Figure 5: Scaling analysis of ESM2 checkpoints on proteome-level PPI retrieval tasks (Recall@K%). Performance remains largely unchanged when increasing model size.

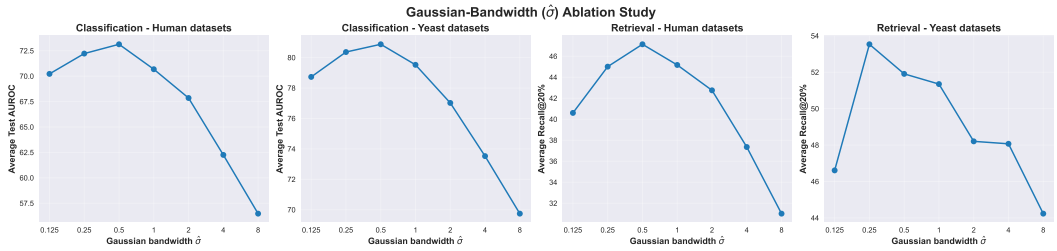

Figure 6: Gaussian-bandwidth ablation: We vary the kernel width $\hat{\sigma}$ from 0.125 to 8 (log-scale x-axis) and report the averaged AUROC/Recall@20% across human and yeast datasets.

bandwidths degrade performance. For example, tiny $\hat{\sigma}$ encourages sparse interactions, yet leads to a clear drop in performance when the bandwidth is too small; very large $\hat{\sigma}$ over-smooths the residue-level structure and significantly hurts both PPI classification and retrieval performance. As $\hat{\sigma} = 0.5$ consistently achieves good performance across species, we adopt this value for all reported experiments.

B.3 Ablation on Low-Rank Attention

Section 3.2 defines rank-r attention as r independent softmax pools (columns sum to one), and Eq. 4 combines the pooled descriptors with the Gaussian feature map in Eq. 3 . For completeness, we derive the rank-r retrieval formulation implied by that construction. Let $\boldsymbol{\Psi}_A \in \mathbb{R}^{L_A \times 2d'}$ and $\boldsymbol{\Psi}_B \in \mathbb{R}^{L_B \times 2d'}$ stack the Random Fourier Features for proteins A and B, and let the attention matrices be $\boldsymbol{W}_A \in \mathbb{R}^{L_A \times r}$ and $\boldsymbol{W}_B \in \mathbb{R}^{L_B \times r}$ with columns that sum to one. The pooled embeddings for each rank are the rows of

$$\boldsymbol{H}_A = \boldsymbol{W}_A^\top \boldsymbol{\Psi}_A \in \mathbb{R}^{r \times 2d'}, \qquad \boldsymbol{H}_B = \boldsymbol{W}_B^\top \boldsymbol{\Psi}_B \in \mathbb{R}^{r \times 2d'}, \tag{15}$$

and the proteome-scale logit becomes

$$\ell(A, B) = \sum_{t=1}^{r} \left\langle \boldsymbol{h}_A^{(t)}, \boldsymbol{h}_B^{(t)} \right\rangle = \operatorname{tr}\left(\boldsymbol{H}_A \boldsymbol{H}_B^\top\right) = \left\langle \operatorname{vec}(\boldsymbol{H}_A), \operatorname{vec}(\boldsymbol{H}_B) \right\rangle, \tag{16}$$

where $\operatorname{vec}(\cdot)$ denotes vectorization. Computing retrieval scores for $r > 1$ therefore amounts to concatenating the r heads into a single embedding of dimension $2d'r$ per protein. Higher ranks can provide stronger expressiveness, but the embedding dimension, memory footprint, and dot-product

Table 4: Impact of attention rank on AUROC and retrieval (macro-average across seven benchmarks). Higher is better; bold marks the best value and underline the second best per metric.

Rank	AUROC (%)	Recall@1%	Recall@3%	Recall@5%	Recall@10%	Recall@20%
1	**75.29**	<u>10.89</u>	<u>18.19</u>	<u>23.31</u>	**33.95**	**48.33**
2	<u>74.56</u>	**11.33**	**18.89**	**24.28**	<u>33.72</u>	<u>47.24</u>
4	69.49	10.23	17.12	22.10	31.47	43.89
8	66.13	8.00	13.95	18.65	27.84	40.39
16	63.86	7.08	12.86	17.16	25.50	37.91
32	64.71	7.46	12.98	17.37	25.76	38.19

cost all grow linearly with r because each additional head contributes another $2d'$ Random Fourier Features (d' for each sin/cos feature).

Table 4 reports the seven-dataset macro-average. Rank 1 already achieves the best AUROC (75.29) and Recall@20% (48.33). Moving to rank 2 mildly improves the very top of the ranking—Recall@1%, Recall@3%, and Recall@5% increase by less than 1% without enhancing AUROC, indicating that the extra head enables slightly stronger early recall capabilities. Larger ranks, however, consistently overfit: AUROC drops below 70 at rank 4 and to 63.86 at rank 16, while Recall@20% degrades from 48.33 (rank 1) to 37.91 (rank 16) despite the $16\times$ increase in memory footprint. Rank 32 further underperforms on every metric. Since rank-one attention is effective enough while achieving the best efficiency compared to higher ranks, we keep $r = 1$ as the default.

B.4 Ablation on Adaptive Negative Weighting Loss

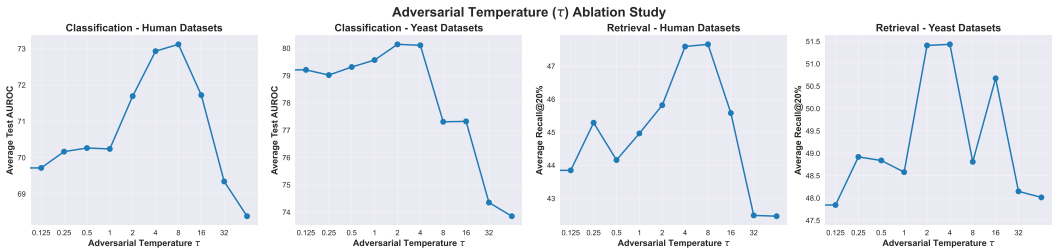

Figure 7: **Adversarial–temperature ablation.** We vary the temperature τ in the adaptive negative weighting loss (Eq. 14) from 0.125 to 32 (log-scale x-axis). Each panel reports the average metric over the indicated datasets. A moderate value of $\tau = 4$ (vertical peak) consistently maximizes both classification AUROC (left two panels) and retrieval Recall@20% (right two panels) on human and yeast benchmarks.

Sensitivity Analysis of τ. As introduced in § 3.4 , we adopt adaptive negative weighting to mitigate the challenge of constructed negatives in PPI datasets. We use a weighted BCE loss that softly prioritizes harder negatives through a temperature parameter (Eq. 13). When $\tau \to 0$, the objective reduces to uniform BCE; when $\tau \to \infty$, it focuses entirely on the single hardest negative in the batch. Figure 7 shows that neither extreme is optimal. Across species and protocols, performance peaks around $\tau = 4$: increasing τ from 0.125 to 4 improves average test AUROC by $\approx$ 2–3 points and Recall@20% by ≈ 3 points, while larger values degrade results by overfitting to outliers. This ablation highlights that moderately emphasizing harder negatives can improve proteome-scale retrieval.

Comparison to Focal Loss In real-world PPI data, samples can vary substantially in difficulty and reliability. Therefore, we propose reweighting them rather than treating all positives and negatives equally. Focal Loss (Lin et al., 2017) was originally proposed for addressing the problem of class imbalance in standard classification settings, where it modulates and balances positive and negative samples using the (α) parameter and further obtains different sample weights based on the $(1-p)^{\gamma}$ term. Our adaptive negative sample weighting, inspired by Sun et al. (2019) , comes from the knowledge graph reasoning (KGR) domain, which typically features (1) sparse ground truth ***pairs***

Model	Test AUROC	Recall@20%
RaftPPI-AdaptiveNegLoss	**75.29**	**49.34**
RaftPPI-FocalLoss	74.16	47.60
RaftPPI-BCE	72.24	45.05

Table 5: **Adaptive weighting vs. focal loss.** Macro-average AUROC and Recall@20 % (means over the seven PPI benchmarks in Table 1).

and (2) ranking-style objectives where the observed positive pair is encouraged to have a higher score compared with many sampled (unreliable) negative pairs. In this setting, it is natural to only reweight negative samples based on their relative hardness within a batch. We view our PPI retrieval scenario as more closely aligned with the KGR setting (as PPI retrieval aims to find the interacting pairs in the proteome and negative PPIs are often constructed as pseudo-negatives by sampling) than with conventional CV classification. Performance-wise, both adaptive negative loss and Focal Loss improve over standard BCE, indicating that PPI samples indeed have different effective quality, and our adaptive negative loss is slightly better on average.

B.5 Ablation on Random Fourier Features Dimension

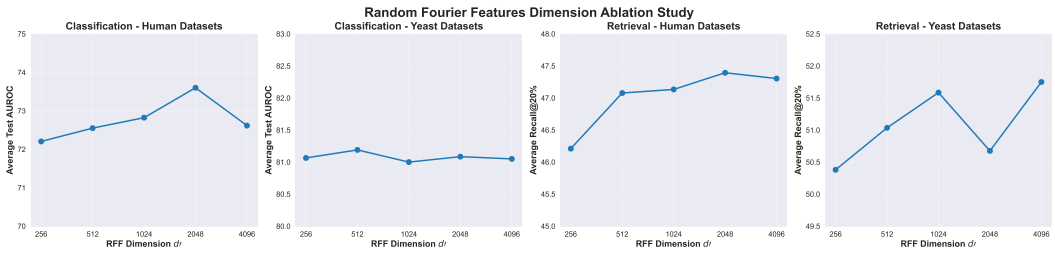

Figure 8: **Random Fourier Features dimension ablation.** We vary the RFF embedding dimension d' from 256 to 4096 (log-scale x-axis; the corresponding sin/cos feature dimension is $2d'$) and report the averaged AUROC/Recall@20% across human and yeast datasets. In contrast to other hyperparameters, performance remains remarkably stable across all tested dimensions, with changes of less than 1.1 points, demonstrating that d' is not an important parameter.

The Random Fourier Features (RFF) embedding dimension d' (where d' is the number of frequencies in Eq. 8) controls the expressiveness of the Gaussian kernel approximation. Higher dimensions provide more accurate kernel approximation but increase computational cost and memory footprint linearly with the embedding dimension (the sin/cos feature dimension scales as $\mathcal{O}(2d')$ and storage scales as $\mathcal{O}(Nd')$ for N proteins). Figure 8 sweeps $d' \in \{256, 512, 1024, 2048, 4096\}$ and averages the metrics within human and yeast datasets.

Compared to the previous hyperparameter ablations, the RFF dimension d' is *not* a sensitive parameter within our tested range. Performance remains remarkably stable across all tested dimensions: varying d' from 256 to 4096 changes average AUROC by less than 1.1 points and Recall@20% by less than 1.1 points on human datasets, with similarly small variations on yeast datasets. These variations are substantially smaller than the change in performance when varying $\hat{\sigma}$ (Appendix B.2) and varying τ (Appendix B.4): for example, on human datasets, $\hat{\sigma} = 0.5$ achieves 73.1 AUROC while $\hat{\sigma} = 8.0$ drops to 56.5 AUROC, a 16.7 point difference. The stability of RFF dimension across a $16\times$ range demonstrates that RaftPPI is robust to the RFF dimension choice, and any reasonable value (e.g., 512–2048) works well in practice.

B.6 Evaluation on PiNUI Datasets

We additionally evaluate our method on the PiNUI datasets (Dubourg-Felonneau et al., 2023) , which provide additional large-scale PPI classification benchmarks for human and yeast. After cleaning, PiNUI-human contains 684,448 protein pairs (228,317 positive, 456,131 negative) and PiNUI-yeast

Dataset	Species	Train			Val			Test			Total
		Pos	Neg	Total	Pos	Neg	Total	Pos	Neg	Total	
PiNUI-human	Human	136,812	273,858	410,670	45,684	91,205	136,889	45,821	91,068	136,889	684,448
PiNUI-yeast	Yeast	31,797	62,424	94,221	10,569	20,838	31,407	10,671	20,736	31,407	157,035

Table 6: Dataset statistics for PiNUI-human and PiNUI-yeast.

Model	PiNUI-human		PiNUI-yeast	
	AUROC	AUPRC	AUROC	AUPRC
ESM2-NoFT	59.14 ± 0.00	42.45 ± 0.00	51.90 ± 0.00	38.74 ± 0.00
ESM2-MLP	75.37 ± 0.66	61.97 ± 1.03	$\underline{77.52 \pm 0.76}$	$\underline{63.01 \pm 0.81}$
PLM-Interact	$\underline{76.04 \pm 0.18}$	$\underline{62.71 \pm 0.35}$	76.77 ± 0.96	61.97 ± 1.71
TUnA	64.53 ± 0.34	47.31 ± 0.55	69.79 ± 1.08	54.29 ± 0.67
RaftPPI-P	73.61 ± 0.31	61.06 ± 0.31	72.53 ± 0.62	58.64 ± 0.67
RaftPPI	$\mathbf{77.92 \pm 0.39}$	$\mathbf{69.33 \pm 0.34}$	$\mathbf{77.87 \pm 0.41}$	$\mathbf{69.24 \pm 0.47}$

Table 7: Test AUROC and AUPRC performance (%) on PiNUI datasets.

contains 157,035 pairs (53,037 positive, 103,998 negative). Both datasets are split randomly into train/validation/test sets with a 60/20/20 ratio; Table 6 summarizes these statistics.

Table 7 reports Test AUROC and AUPRC results across competing methods. The unsupervised ESM2 baseline, i.e., ESM2-NoFT, performs poorly on both datasets, highlighting the importance of fine-tuning for PPI prediction. RaftPPI achieves the best performance on both datasets. These results are consistent with our findings on the seven-dataset benchmark discussed in § 4.2 , demonstrating that the proposed residue-level interaction modeling and adaptive negative weighting in RaftPPI generalize well to larger-scale PPI datasets. PLM-Interact and ESM2-MLP appear to also have strong performance. Besides, the improvements of RaftPPI v.s. RaftPPI-P (protein-level interaction only) further demonstrate the effectiveness of our design that considers residue-level interaction.

C Use of Large Language Models

We used a large language model to help polish the writing. We take full responsibility for all content in this paper.