CQFINER: PROTEIN CONTEXT QUERY REFINEMENT FOR TEST-TIME RESOLUTION SCALING IN BINDING SITE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Protein binding site prediction is one of the most fundamental problems in drug discovery and structural biology. A mainstream approach is to place anchors or query nodes at different locations, move and classify these queries, and finally use the coordinates of high-confidence queries as the predicted binding site locations. Resolution is a key determinant of predictive performance of such methods. However, in this work we demonstrate that achievable spatial resolution is constrained by GPU memory, over-smoothing, and suboptimal training dynamics. More importantly, these query-based methods fix the query number and distribution during training and inference, thus cannot benefit from higher spatial resolution at inference, or tailored query sampling according to user priors. Here we introduce Context Query Refiner (CQFiner). CQFiner models the protein as a fixed read-only context and formulates binding site prediction as an object detection task conditioned on this context, refining both query positions and confidence scores from coarse to fine. We demonstrate how object detection practices help the model benefit from higher resolution and how read-only protein context enables test-time scaling and prior-based query sampling. Based on the architecture, CQFiner substantially outperforms state-of-the-art methods on both in-distribution and out-of-distribution benchmarks, including COACH420, HOLO4K, etc. Code and model weights are available at `https://anonymous.4open.science/r/CQFiner-E787/README.md`.

1 INTRODUCTION

Protein binding site prediction (BSP) is a fundamental problem in structural biology and drug discovery (Nakashima et al., 2011; Vincent et al., 2022; Chan et al., 2019). Because ligand-binding pockets largely determine protein function and druggability (Fan et al., 2025), accurate BSP provides a strong spatial prior for downstream applications such as molecular docking, virtual screening, de novo ligand design, and functional annotation.

Recently, deep-learning-based methods have demonstrated higher accuracy than traditional rule- or template-based approaches (Fan et al., 2025). Existing methods mainly follow two paradigms. (1) Classification-based methods predict binding-site membership for residues, atoms, or voxels, then cluster positive predictions to obtain pocket centers (Smith et al., 2024; Fan et al., 2025; Carbery et al., 2024; Kandel et al., 2021). (2) Query-based methods place candidate nodes at potential pocket locations and predict confidence scores (Li et al., 2023). Offset-based variants further move these nodes toward nearby pocket centers (Sestak et al., 2025). Such variants directly supervise center coordinates and commonly achieve stronger pocket localization (Fan et al., 2025).

For query-based methods, spatial resolution is a key determinant of predictive performance. More and denser queries provide better coverage of the protein and increase the likelihood that each binding site has a nearby candidate, ultimately improving performance. However, in this work, we demonstrate that achievable spatial resolution is constrained by GPU memory, over-smoothing, and training dynamics. At excessively high query resolution, the dense connectivity between queries and the protein increases memory consumption and can make query representations increasingly similar. Meanwhile, those "hard" queries (i.e., far from the target binding site) can dominate optimization

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

due to larger loss despite their limited contribution to accurate localization, thus resulting in suboptimal performance. On the other hand, the available resolution, and even the distribution of queries, is fixed during training. At test time, using a different query distribution or number of queries may introduce out-of-distribution (OOD) inputs, leading to a sharp degradation in performance. As a result, the model cannot scale to higher resolutions at inference time or adapt the query distribution according to user-provided priors to achieve better performance. These constraints substantially limit both the performance ceiling and practical applicability of query-based methods.

To address these limitations, we introduce Context Query Refiner (CQFiner). The central idea is to treat the protein as a fixed context and formulate binding site prediction as an object detection task conditioned on this context, refining both query positions and confidence scores. Specifically, CQFiner adopts a coarse-to-fine query refinement strategy. An expressive graph neural network (GNN) backbone first extracts protein features and performs an initial displacement of query positions. These protein features then serve as a fixed context for a lightweight EGNN, which further refines query positions and confidence scores while supporting layerwise coordinate updates. We show how object detection principles mitigate over-smoothing and reduce computational complexity, while a read-only context enables test-time scaling and prior-based query sampling. CQFiner substantially outperforms state-of-the-art methods on both in-distribution and out-of-distribution benchmarks, including COACH420, HOLO4K, APOBind, and LIGYSIS.

2 MOTIVATION

We use a minimal baseline to analyze how spatial resolution affects BSP. Related work is provided in Appendix A.

Experimental Settings: We represent each protein as a residue graph (Sestak et al., 2025; Wang & Dokholyan, 2026) and sample query nodes around it, initialized with spatial coordinates and learnable embeddings. A GNN processes the protein–query graph, whose edges follow a predefined distance cutoff, and predicts each query's probability of having a binding site nearby. Non-maximum suppression (NMS) selects high-confidence query coordinates as the final pocket centers.

Metrics: Distance center–center (DCC@4) and distance center–atom (DCA@4) measure the fraction of ground-truth sites recovered by the top-n predictions, where n is the number of sites in each protein. A site is recovered if a predicted center lies within 4 Å of its ligand center (DCC) or any ligand atom (DCA). AP-IoU evaluates residue masks from an object detection perspective: confidence-ranked predictions are matched to ground-truth masks at a specified intersection-over-union (IoU) threshold to compute average precision (Fan et al., 2025).

2.1 HOW SPATIAL RESOLUTION AFFECTS MODEL ACCURACY

We first investigate the effect of spatial resolution on model performance. By varying the number of queries, we can directly control the spatial resolution: more queries provide more complete coverage of the protein surface and increase the likelihood that the model can identify queries located near true pockets. Fig. 1 shows the relationship between DCC and the number of queries. The trend is consistent and clear: increasing the number of queries leads to higher spatial resolution and better model performance. This result is intuitive: as the number of query nodes increases, each site is more likely to be covered by multiple queries, which increases the probability that at least one query is classified as positive and makes the prediction more robust. These results demonstrate the importance of spatial resolution for BSP.

2.2 ENHANCING RESOLUTION VIA OFFSET PREDICTION

Can we achieve higher spatial resolution without directly increasing the number of queries? Fundamentally, binding site prediction resembles an object detection task on the protein surface, where the role of a query is analogous to a region proposal in R-CNN (Girshick et al., 2014). Such models require a large number of proposals to ensure that no target is missed. In later one-stage object detection frameworks such as YOLO (Redmon et al., 2016), however, proposals are effectively replaced by a fixed initial grid, and the actual object bounding boxes are obtained through offset prediction. The same idea can be adopted for BSP. The initially sampled queries can be regarded as a grid, and

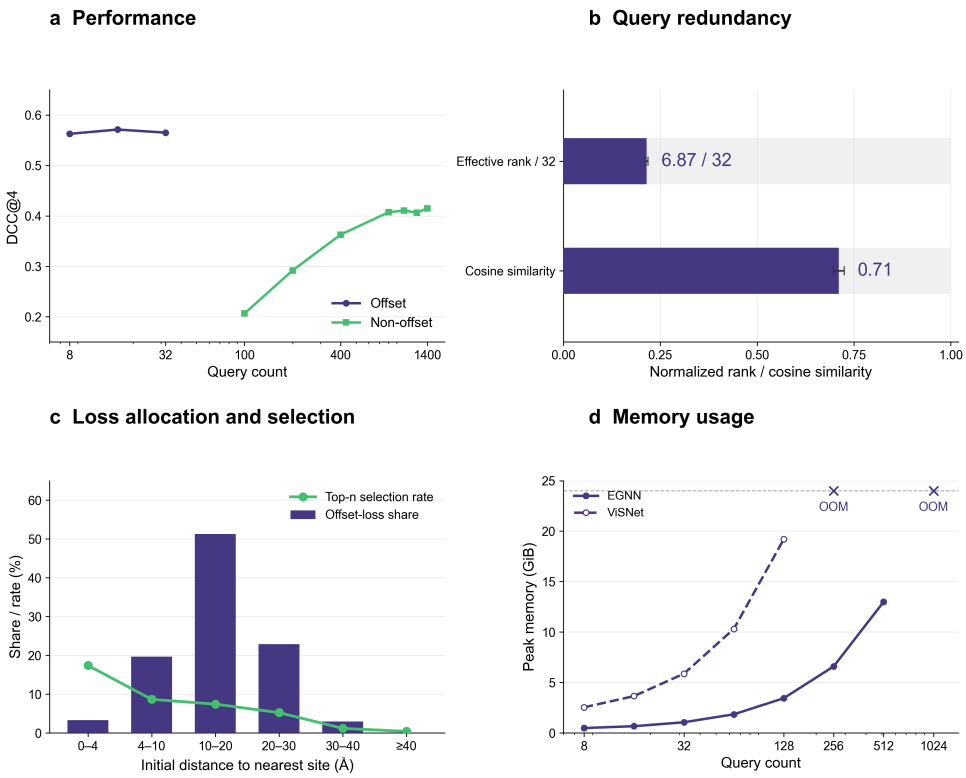

Figure 1: **Spatial resolution and limitations of offset-based prediction. a. Performance.** UniSite validation DCC@4 versus query count for non-offset and offset baselines. **b. Query over-smoothing.** Final-layer effective rank (6.87/32) and mean pairwise cosine similarity (0.71) for the fully connected baseline on 255 validation proteins; error bars show 95% batch-bootstrap confidence intervals. **c. Loss allocation and selection.** Epoch-35 offset-loss share and post-NMS top-n selection rate by initial nearest-site distance (256 training proteins). Loss share uses total loss as denominator; selection rate uses the query count within each bin. **d. Memory.** Peak allocated memory of fully connected EGNN and ViSNet allocation probes on three proteins near the 90th percentile of residue count (batch size 1, mixed precision). Lines/bands show medians/ranges; OOM means at least one protein exceeds GPU capacity (Appendix D).

we directly predict the offset from each initial query to the nearest pocket center. Such method was possibly first proposed by VN-EGNN (Sestak et al., 2025).

In principle, this approach offers arbitrarily fine spatial resolution, as queries can move to any location, with localization precision determined by the model's predictive accuracy. Following VN-EGNN, we add offset prediction to the baseline. This yields a substantial improvement: the VN-EGNN-style model requires only 8 queries to outperform the original baseline with 1,400 queries. These results highlight the importance of offset prediction for query-based BSP.

A natural question is whether combining offset prediction with denser query sampling yields further improvements. In practice, however, as shown in Fig. 1a, increasing the number of queries from 8 to 32 leads to unstable and modest changes in performance. This phenomenon is primarily due to over-smoothing and training dynamics, which we examine in the next section.

2.3 What Limits the Benefits of More Queries?

In summary, the reason why VN-EGNN-style models fail to benefit from higher spatial resolution stems from the fully connected residue–query graph, which exacerbates over-smoothing, creates optimization difficulties through imbalanced loss contributions, and increases memory consumption. Because the eventual binding site of each query is unknown in advance and may be nearby or far

away, VN-EGNN connects every residue to every query. This rationale is similar to that underlying the use of fully connected graphs in many molecular generation models. Such practice results in the following drawbacks.

Over-smoothing. Because every query receives messages from the same complete set of residues, full connectivity exacerbates over-smoothing. To illustrate this effect, Fig. 1b demonstrates the effective rank and similarity of queries. Full connectivity produces query representations with large inter-query similarity and an effective rank of only 6.87 across 32 queries. This suggests that increasing the number of queries may not yield additional informative features.

Training Dynamics. The training dynamics introduce another difficulty. As shown in Fig. 1c, as the distance between a query and its assigned target increases, the final selection rate is lower. However, such queries account for large loss contributions. The model wastes capacity on these "hard" queries, which are less useful for site localization, and thus may result in degraded performance.

Memory consumption. Even setting these issues aside, scaling the number of queries remains difficult: full residue–query connectivity introduces a large number of edges and substantially increases GPU memory consumption. As shown in Fig. 1d, the ViSNet (Wang et al., 2024b) allocation probe uses 19.20 GiB at 128 queries and reaches OOM at 256 on a 24 GB GPU. This imposes a restrictive practical limit on the number of queries.

3 METHOD

In this section, we first introduce the design choices that address the above limitations, then present the overall pipeline of CQFiner.

3.1 INCREASING THE UPPER LIMIT OF RESOLUTION VIA OBJECT DETECTION PRACTICES

As discussed above, BSP can be viewed as an **object detection task**, analogous to those addressed by YOLO and VoteNet. YOLO combines object confidence prediction with bounding box regression, corresponding to confidence and offset prediction in BSP. Unlike the VN-EGNN-style formulation, it predicts confidence across all grid cells but restricts regression to cells assigned to objects. This suggests a solution to the training dynamics and over-smoothing issues: supervise confidence for all queries but offsets only for those near binding sites. In this case, first, these queries no longer require connections to every residue to support long-range localization, allowing local, cutoff-based connectivity instead. Such graph thus reduces feature homogenization and over-smoothing. Second, distant queries no longer contribute to the offset loss and cannot dominate optimization, allowing the network to focus on refining nearby queries. Finally, local connectivity reduces edge counts, lowering memory consumption. With these techniques, the model can benefit from higher-resolution queries.

3.2 PROTEIN READ-ONLY CONTEXT FOR TEST-TIME SCALING AND PRIOR INJECTION

Even if the model itself can benefit from higher resolution, the resolution available during training is inevitably constrained by factors such as GPU memory. Meanwhile, the query distribution at inference time is typically kept consistent with that used during training to avoid out-of-distribution (OOD) inputs. We therefore seek to decouple the resolution used during training from that used at inference time.

CQFiner achieves this by **treating the protein as a read-only context and enforcing independence among queries**. Specifically, CQFiner removes all outgoing edges from queries, retaining only residue–residue edges and residue-to-query edges. Under this design, queries are mutually independent and cannot affect the protein representations. Consequently, adding or removing queries at inference time does not alter the representations of other queries or residues. This design provides two key benefits. (1) **Test-time scaling**: the model can use a higher query resolution at inference time than during training, thereby achieving improved performance. (2) **Prior injection**: users can sample queries according to domain-specific priors, for example by allocating queries more densely to regions that are more likely to contain binding sites, thereby improving either prediction accuracy or computational efficiency.

3.3 COARSE-TO-FINE QUERY REFINEMENT

The cutoff-based residue–residue and directed residue-to-query graph reduces memory consumption, making expressive backbones such as ViSNet practical for BSP. However, their feature extraction mechanisms are not directly compatible with iterative query movement. Modern GNNs such as ViSNet maintain edge features across layers, whereas moving queries changes their geometry relative to neighboring residues, potentially making the retained edge features inconsistent with the updated coordinates. The challenge is therefore to exploit expressive geometric representations while preserving the flexibility of layerwise coordinate refinement.

We address this challenge with a **Coarse-to-Fine** strategy, using the read-only protein context to separate feature extraction from coordinate refinement. In the coarse stage, ViSNet extracts residue and query features with query coordinates fixed throughout its layers, then predicts a one-shot displacement for each query. In the fine stage, a lightweight EGNN reuses the extracted residue features as fixed context and updates query features and coordinates through directed residue-to-query messages. Unlike ViSNet, EGNN recomputes geometric edge features from the current coordinates at each layer, allowing iterative refinement before producing the final positions and confidence scores. This design preserves geometric consistency during feature extraction and corrects coarse localization errors without repeatedly running the more expensive backbone.

3.4 RESIDUE-LEVEL MASK PREDICTION

Site-center localization alone does not identify which residues belong to each predicted site. We therefore use the query representation to predict a site-specific residue mask through a distance-gated dot-product head. Residue and query features are projected into a shared space, modulated by a distance-dependent gate, and combined to predict the probability that residue i belongs to the site represented by query j:

$$\hat{y}_{ij} = \sigma\Big(D^{-1/2}\mathbf{1}^\top(\phi_h(\mathbf{h}_i) \odot \phi_q(\mathbf{q}_j) \odot g(\mathrm{RBF}(d_{ij}/r_j)))\Big), \qquad d_{ij} = \left\|\mathbf{x}_i - \mathbf{x}_j^{(q)}\right\|_2. \quad (1)$$

Here, $\mathbf{h}_i$ and $\mathbf{q}_j$ are residue and query features; ϕ_h and ϕ_q are multi-layer perceptrons (MLPs) with output dimension D; $g = 2\sigma(\mathrm{MLP}(\cdot))$ is the distance gate; and r_j is a query-conditioned distance scale. Here $\odot$ denotes element-wise multiplication, RBF is a radial basis expansion, and σ is the sigmoid function (Appendix F). The distance gate reflects the model's locality: each query receives messages only from nearby residues and should not be expected to reliably classify distant residues.

3.5 THE OVERALL PIPELINE OF CQFINER

As shown in Fig. 2, the overall pipeline of CQFiner consists of the following components.

Dense sampling: We identify surface atoms using solvent-accessible surface area (SASA), sample queries randomly during training, and use farthest point sampling for default inference. Queries are displaced outward along local surface normals estimated by principal component analysis (PCA). This strategy maintains sufficient spatial resolution while avoiding the computational cost of an excessively large query set.

Graph construction and message passing: We construct a cutoff-based protein–query graph containing residue–residue and directed residue-to-query edges. Query-to-residue and query-to-query edges are excluded, so queries receive local protein information without modifying the protein context.

Coarse query displacement: ViSNet (Wang et al., 2024b) extracts residue and query features with query coordinates fixed across its layers. An EGNN-style displacement head then uses query and residue-to-query edge features to predict an equivariant offset, supervised toward nearby site centers. Coordinates are updated only after feature extraction to preserve geometric consistency.

Fine-grained refinement: After applying the coarse displacements, we construct a graph with only directed residue-to-query edges and reuse the encoded residue features as fixed context. EGNN (Satorras et al., 2021) then updates query features and coordinates layer by layer to obtain the final positions and confidence scores.

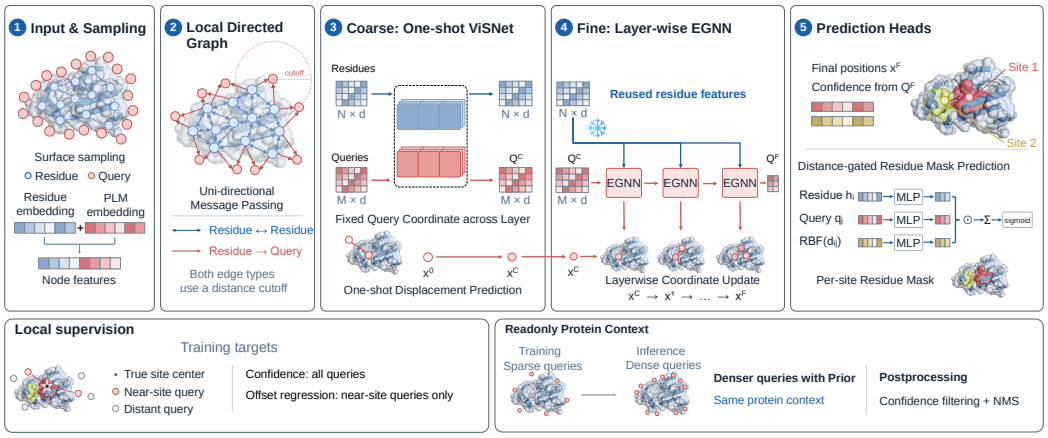

Figure 2: **Overview of CQFiner. (1–2)** Surface-sampled queries and residues form a local graph with residue–residue and directed residue-to-query edges. **(3)** ViSNet extracts features with fixed query coordinates, then predicts coarse displacements. **(4)** EGNN refines queries layerwise against fixed residue features. **(5)** Queries predict site centers, confidence scores, and distance-gated residue masks. **Bottom:** Confidence supervision covers all queries; offset supervision covers near-site queries. Query-independent protein context supports test-time scaling and prior injection. Confidence filtering and NMS select final predictions.

Post-processing: Confidence-ranked query centers undergo NMS for localization evaluation. For mask evaluation, the distance-gated head predicts residue masks before a separate NMS grouping step fuses masks from nearby queries (Appendix F).

Table 1: **UniSite validation-set AP and main test-set results on HOLO4K and COACH420.** $AP_{0.3}$ and $AP_{0.5}$ denote AP under mask IoU thresholds 0.3 and 0.5. DCC and DCA are top-n success rates at 4 Å. The best results are bolded and the second-best results are underlined; values within 0.010 are treated as tied.

		UniSite val.		HOLO4K			COACH420		
Method	Type	$AP_{0.3}$	$AP_{0.5}$	$AP_{0.3}$	DCC	DCA	$AP_{0.3}$	DCC	DCA
Fpocket	Geometry	0.184	0.102	0.271	0.308	0.438	0.211	0.271	0.411
Fpocket-rescore	ML	0.508	0.235	0.590	0.518	0.765	0.560	0.440	0.711
P2Rank	ML	0.506	0.216	0.601	0.530	<u>0.819</u>	0.619	0.464	0.741
DeepPocket	CNN	0.427	0.233	0.542	0.492	0.737	0.518	0.396	0.676
GrASP	GNN	0.447	0.285	0.667	0.513	0.742	<u>0.715</u>	0.485	<u>0.762</u>
VN-EGNN	GNN	0.162	0.071	0.261	<u>0.586</u>	0.700	0.264	<u>0.545</u>	0.753
UniSite-1D	Transformer	0.512	0.303	0.687	0.554	0.769	0.592	0.455	0.735
UniSite-3D	Transformer	<u>0.560</u>	<u>0.384</u>	<u>0.709</u>	0.572	0.788	<u>0.720</u>	0.470	0.738
CQFiner	GNN	**0.657**	**0.502**	**0.805**	**0.767**	**0.877**	**0.816**	**0.735**	**0.860**

4 Experiment

4.1 Overall results of CQFiner

In Table 1, we first report the performance of CQFiner on the UniSite benchmark, which includes a validation set together with the HOLO4K and COACH420 test sets. We follow the default UniSite split and report validation-set AP together with main test-set AP-IoU, DCC, and DCA, as in the original paper (Fan et al., 2025). Across these metrics, CQFiner demonstrates substantially stronger performance than previous methods. On both HOLO4K and COACH420, the DCC scores are improved by more than 15 percentage points compared with the best previous methods.

4.2 Robustness over out-of-distribution datasets

Table 2: **Performance on structural-robustness and OOD benchmarks.** $AP_{0.3}$ and $AP_{0.5}$ use residue-mask IoU thresholds of 0.3 and 0.5; DCC and DCA are top-n success rates at 4 Å. All CQFiner results use the same UniSite-validation-selected checkpoint and 200 surface queries, without OOD-specific tuning. LIGYSIS uses site-center proxy atoms, so DCA equals DCC. Best results are bold and second-best results are underlined within each dataset and metric; ties are determined at the reported precision.

		VN-EGNN				CQFiner				UniSite			
Dataset	Structure	$AP_{0.3}$	$AP_{0.5}$	DCC	DCA	$AP_{0.3}$	$AP_{0.5}$	DCC	DCA	$AP_{0.3}$	$AP_{0.5}$	DCC	DCA
COACH420	Experimental	0.264	0.214	0.545	0.753	**0.816**	**0.634**	**0.735**	**0.860**	0.720	0.566	0.470	0.738
HOLO4K	Experimental	0.261	0.186	0.586	0.700	**0.805**	**0.640**	**0.767**	**0.877**	0.709	0.591	0.572	0.788
COACH420	AlphaFold	0.244	0.154	0.458	0.694	**0.767**	**0.536**	**0.678**	**0.837**	0.686	0.512	0.498	0.721
HOLO4K	AlphaFold	0.252	0.156	0.505	0.681	**0.759**	0.575	**0.720**	**0.856**	0.707	**0.592**	0.614	0.757
APOBind	Holo	0.237	0.141	0.455	0.658	**0.564**	**0.234**	**0.571**	**0.722**	0.369	0.130	0.365	0.559
APOBind	Apo	0.229	0.107	0.411	0.609	**0.525**	**0.162**	**0.494**	**0.664**	0.399	0.102	0.334	0.544
LIGYSIS	External	0.074	0.035	0.153	0.153	**0.261**	**0.129**	0.193	0.193	0.203	0.123	**0.215**	**0.215**

COACH420 and HOLO4K predominantly contain ligand-bound (holo) structures, whereas practical BSP often uses unbound (apo) or predicted structures. Recent evaluations have also identified incomplete coverage of binding interfaces across structures and inadequate assessment of ranking and redundancy, motivating benchmarks such as LIGYSIS (Utgés & Barton, 2024). We therefore evaluate CQFiner on APOBind (Aggarwal et al., 2021b), AlphaFold-predicted structures, and LIGYSIS against UniSite and VN-EGNN, strong baselines for AP-IoU and DCC/DCA, respectively.

As shown in Table 2, CQFiner outperforms the competing methods on most datasets and maintains its overall advantage on AlphaFold-predicted structures. The apo and LIGYSIS datasets are substantially more challenging, with all methods exhibiting considerable performance degradation.

4.3 Read-only protein context: test-time scaling and benefits from priors

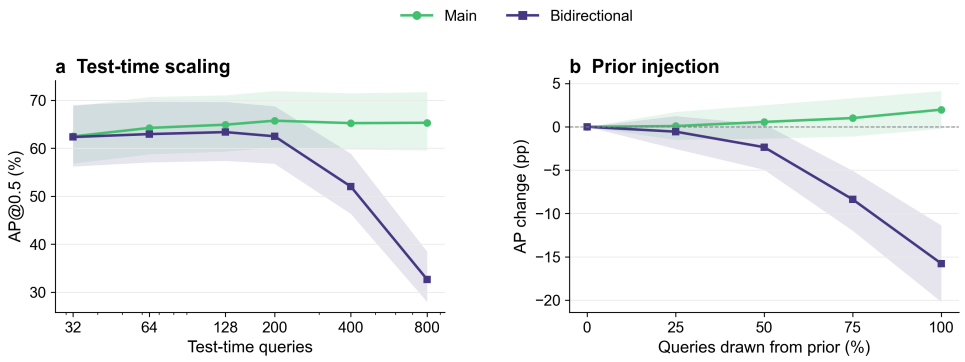

Figure 3: **Test-time query scaling and oracle prior injection. a. Test-time query scaling.** AP at IoU@0.5 versus inference query count for models trained with 32 queries. CQFiner improves and then plateaus, whereas the bidirectional variant deteriorates at large query counts. **b. Oracle prior injection.** AP change when increasing the fraction of queries sampled near known binding sites. The oracle prior uses Gaussian surface-sampling weights centered on ground-truth ligand centers ($\sigma = 6$ Å). CQFiner shows a modest mean gain, while the bidirectional variant exhibits substantial degradation at high prior fractions. Curves average results across COACH420 and HOLO4K (128 proteins each) and three paired query-sampling seeds. Shaded bands show 95% paired protein-bootstrap confidence intervals. Sampling and evaluation protocols are given in Appendix E.

Despite similar $AP_{0.5}$ under the default query configuration (Table 3), bidirectional and unidirectional message passing differ markedly in robustness to query sampling (Fig. 3). For models trained with 32 queries, increasing the inference count from 32 to 800 changes mean $AP_{0.5}$ from 62.45% to 65.30% for CQFiner and from 62.33% to 32.64% for the bidirectional model; CQFiner plateaus around 200 queries.

At fixed training and inference budgets of 200 queries, increasing the oracle-prior fraction from 0% to 100% changes $AP_{0.5}$ by $+1.98$ percentage points for CQFiner and -15.77 percentage points for the bidirectional model. These results indicate that bidirectional message passing is inherently sensitive to shifts in the query distribution, even when such shifts are favorable for prediction, thereby limiting test-time scaling and the incorporation of prior-guided sampling. In contrast, CQFiner's read-only protein context can effectively benefit from both increased test-time resolution and prior-guided query sampling.

Table 3: **Ablation experiments of different components of CQFiner.** "w/o Refine" removes the EGNN stage and uses two ViSNet passes at inference. "w/o Read-only Context" enables bidirectional residue–query messages in both stages. "FullConn + All Queries" replaces local residue–query connectivity in both stages with full connectivity and supervises all queries; the ViSNet cutoff is 60 Å. "Dot → DETR" replaces the mask head, matching objective and mask scoring, using 32 additional learned mask queries. Each cell reports the score and its change relative to CQFiner, computed before rounding; red/green deltas indicate increases/decreases of at least 0.010, and gray deltas indicate smaller changes or the baseline.

Modification Scope	Variant	UniSite val.			HOLO4K			COACH420		
		$AP_{0.5}$	DCC	DCA	$AP_{0.5}$	DCC	DCA	$AP_{0.5}$	DCC	DCA
-	**CQFiner**	0.502 (0.000)	0.604 (0.000)	0.731 (0.000)	0.640 (0.000)	0.767 (0.000)	0.877 (0.000)	0.634 (0.000)	0.735 (0.000)	0.860 (0.000)
Refinement	w/o Refine	0.476 (-0.026)	0.596 (-0.008)	0.727 (-0.003)	0.633 (-0.007)	0.753 (-0.014)	0.875 (-0.002)	0.615 (-0.019)	0.673 (-0.062)	0.842 (-0.018)
Graph Design	w/o Read-only Context	0.507 (+0.005)	0.600 (-0.004)	0.734 (+0.004)	0.654 (+0.015)	0.740 (-0.027)	0.871 (-0.006)	0.642 (+0.008)	0.661 (-0.074)	0.869 (+0.009)
	FullConn + All Queries	0.403 (-0.099)	0.515 (-0.089)	0.659 (-0.071)	0.553 (-0.086)	0.669 (-0.098)	0.823 (-0.054)	0.505 (-0.129)	0.634 (-0.101)	0.830 (-0.030)
Residue Mask	Dot → DETR	0.474 (-0.029)	0.589 (-0.015)	0.732 (+0.002)	0.618 (-0.021)	0.736 (-0.031)	0.859 (-0.018)	0.580 (-0.053)	0.670 (-0.065)	0.884 (+0.024)

4.4 Ablation Studies

Local connectivity and supervision. Inspired by YOLO, local connectivity and supervision avoid unnecessary long-range message passing, reducing over-smoothing and preventing distant queries from dominating offset regression. Replacing them with a fully connected residue–query graph and offset supervision for all queries produces the largest overall performance drop (Table 3), with $AP_{0.5}$ and DCC decreasing by approximately 10 percentage points and DCA also declining. This highlights the importance of refining queries at an appropriate spatial scale rather than allocating model capacity to less useful distant queries.

Coarse-to-fine query refinement. The coarse-only strategy requires two ViSNet passes for its best inference performance: one to predict displacements and another to estimate confidence at the updated positions. Fine-stage refinement avoids repeating the backbone and offers a second advantage: whereas the coarse stage updates protein and query features jointly, the fine stage updates only query features against fixed protein features. This allows query representations to specialize for residue-mask prediction. Table 3 shows improvements in $AP_{0.5}$ across all three splits, while localization gains vary by dataset.

Distance-gated residue mask prediction. Table 3 compares our distance-gated dot-product head with a UniSite-style DETR decoder. Using only a few MLPs and a distance-gated dot product instead of Transformer layers, our head improves AP-IoU by approximately 2–5 percentage points. Distance gating is central to this design: local connectivity limits each query's information about distant residues, making query–residue feature dot products less informative for those residues. The gate suppresses distant residue predictions using spatial proximity, improving mask prediction (Appendix C).

4.5 MECHANISM ANALYSIS OF LOCAL DIRECTED GRAPH

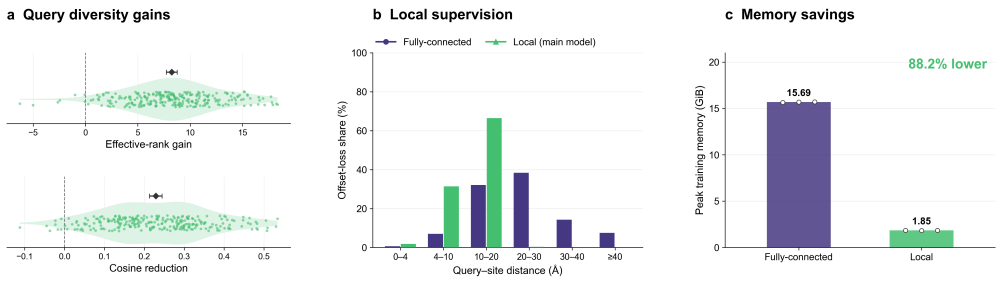

Figure 4: **Query feature diversity, supervision allocation, and memory savings with the local strategy. a. Query diversity gains.** Paired final-layer changes across 256 validation proteins: effective rank (local minus full) and cosine similarity (full minus local). Diamonds and error bars show means and 95% bootstrap confidence intervals. **b. Local supervision.** Distribution of stage-0 offset-loss contributions by initial query-to-site-center distance across 256 training proteins. **c. Memory savings.** At 200 queries, the local model reduces median peak allocated training memory from 15.69 to 1.85 GiB (88.2%). Protocols are detailed in Appendix D.

We examine how local connectivity and supervision address the three limitations in Sec. 2.3.

Over-smoothing. Compared with the fully connected variant, CQFiner's local graph yields substantially higher effective ranks and lower similarity among query representations (Fig. 4a), demonstrating that local connectivity mitigates over-smoothing.

Training Dynamics. In the VN-EGNN-style baseline, distant queries account for most of the offset regression loss. Fig. 4b compares loss contributions across query-to-site distances for the fully connected and local variants of CQFiner. Local supervision concentrates optimization on queries initialized near binding sites, supporting fine-grained refinement. Queries with initial site-center distances of at least 20 Å contribute 60.22% of the stage-0 offset loss in the fully connected variant, compared with 0.34% locally. The supervision cutoff uses ligand-atom distance, whereas the plotted bins use site-center distance (Appendix F).

Memory consumption. Fig. 4c shows GPU memory consumption at 200 queries: 15.69 GiB for full connectivity and 1.85 GiB for local connectivity. Full connectivity incurs substantially higher memory costs, with the first observed OOM at 512 queries on a consumer GPU with 24 GB of memory. In contrast, local connectivity accommodates all 4,096 tested queries within the same memory budget.

5 CONCLUSION

In this work, we introduce CQFiner, a query-based framework for protein binding site prediction that addresses the resolution bottleneck caused by over-smoothing, unfavorable training dynamics, and memory cost. By combining local supervision, directed residue-to-query message passing, coarse-to-fine refinement, and a read-only protein context, CQFiner supports higher-resolution inference and prior-guided query sampling without retraining. Experiments across in-distribution and OOD benchmarks demonstrate strong improvements in binding-site localization and residue-mask prediction, suggesting that query resolution can be treated as a flexible inference-time resource rather than a fixed architectural constraint.

AI USE STATEMENT

We used generative AI tools, including OpenAI Codex, to assist with method implementation and debugging, experimental protocols, analysis and interpretation of results, figure-generation code, literature searches, and manuscript drafting, translation, and editing. AI-assisted code and analyses were checked through execution tests and comparisons with stored experimental outputs and source implementations. The authors directed the research and take responsibility for the final text, claims, results, and associated artifacts.

Ethics statement

CQFiner is expected to improve the localization of protein binding sites, which may benefit structure-based drug discovery, protein function annotation, and the prioritization of downstream experimental validation. Because it is a general computational tool for modeling protein–ligand interaction regions, it may also have dual-use risks: improved binding-site prediction could lower the cost of early-stage screening and could be misused in harmful settings, including the development of illicit or unsafe bioactive compounds. At the same time, CQFiner does not generate ligands, synthesis routes, or actionable experimental protocols by itself, and its outputs still require substantial downstream expertise and validation. We therefore view this work as a foundational modeling contribution and encourage its use only in legitimate biomedical research settings with appropriate legal, ethical, and institutional oversight.

Reproducibility statement

The main benchmark follows the UniSite data splits (Fan et al., 2025). Appendix F specifies data construction, query sampling, model architecture, loss functions, optimization, checkpoint selection, and post-processing, including the training seed and computational settings. Appendix E details the prior-sampling distribution, query budgets, paired evaluation subsets and sampling seeds, and bootstrap procedure used in Fig. 3.

References

Rishal Aggarwal, Akash Gupta, Vineeth Chelur, CV Jawahar, and U Deva Priyakumar. Deeppocket: ligand binding site detection and segmentation using 3d convolutional neural networks. *Journal of Chemical Information and Modeling*, 62(21):5069–5079, 2021a.

Rishal Aggarwal, Akash Gupta, and U Priyakumar. Apobind: a dataset of ligand unbound protein conformations for machine learning applications in de novo drug design. *arXiv preprint arXiv:2108.09926*, 2021b.

Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in neural information processing systems*, 35:11423–11436, 2022.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1): 2453, 2022.

Anna Carbery, Martin Buttenschoen, Rachael Skyner, Frank von Delft, and Charlotte M Deane. Learnt representations of proteins can be used for accurate prediction of small molecule binding sites on experimentally determined and predicted protein structures. *Journal of Cheminformatics*, 16(1):32, 2024.

HC Stephen Chan, Yi Li, Thamani Dahoun, Horst Vogel, and Shuguang Yuan. New binding sites, new opportunities for gpcr drug discovery. *Trends in biochemical sciences*, 44(4):312–330, 2019.

Jigang Fan, Quanlin Wu, Shengjie Luo, and Liwei Wang. Unisite: The first cross-structure dataset and learning framework for end-to-end ligand binding site detection. *arXiv preprint arXiv:2506.03237*, 2025.

Yitian Fang, Yi Jiang, Leyi Wei, Qin Ma, Zhixiang Ren, Qianmu Yuan, and Dong-Qing Wei. Deepprosite: structure-aware protein binding site prediction using esmfold and pretrained language model. *Bioinformatics*, 39(12):btad718, 2023.

Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.

Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C Lawrence Zitnick, and Abhishek Das. Gemnet-oc: developing graph neural networks for large and diverse molecular simulation datasets. *arXiv preprint arXiv:2204.02782*, 2022.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

José Jiménez, Stefan Doerr, Gerard Martínez-Rosell, Alexander S Rose, and Gianni De Fabritiis. Deepsite: protein-binding site predictor using 3d-convolutional neural networks. *Bioinformatics*, 33(19):3036–3042, 2017.

Jeevan Kandel, Hilal Tayara, and Kil To Chong. Puresnet: prediction of protein-ligand binding sites using deep residual neural network. *Journal of cheminformatics*, 13(1):65, 2021.

Radoslav Krivák and David Hoksza. P2rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure. *Journal of cheminformatics*, 10(1):39, 2018.

Vincent Le Guilloux, Peter Schmidtke, and Pierre Tuffery. Fpocket: an open source platform for ligand pocket detection. *BMC bioinformatics*, 10(1):168, 2009.

Shuya Li, Tingzhong Tian, Ziting Zhang, Ziheng Zou, Dan Zhao, and Jianyang Zeng. Pocketanchor: Learning structure-based pocket representations for protein-ligand interaction prediction. *Cell Systems*, 14(8):692–705, 2023.

Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in neural information processing systems*, 33:21002–21012, 2020.

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*, pp. 2980–2988, 2017.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pp. 21–37. Springer, 2016.

Ryosuke Nakashima, Keisuke Sakurai, Seiji Yamasaki, Kunihiko Nishino, and Akihito Yamaguchi. Structures of the multidrug exporter acrb reveal a proximal multisite drug-binding pocket. *Nature*, 480(7378):565–569, 2011.

Qizhi Pei, Kaiyuan Gao, Lijun Wu, Jinhua Zhu, Yingce Xia, Shufang Xie, Tao Qin, Kun He, Tie-Yan Liu, and Rui Yan. Fabind: Fast and accurate protein-ligand binding. *Advances in Neural Information Processing Systems*, 36:55963–55980, 2023.

Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9277–9286, 2019.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. Yolo26: key architectural enhancements and performance benchmarking for real-time object detection. *arXiv preprint arXiv:2509.25164*, 2025.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pp. 9323–9332. PMLR, 2021.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.

Kristof Schütt, Oliver Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, pp. 9377–9388. PMLR, 2021.

Florian Sestak, Lisa Schneckenreiter, Johannes Brandstetter, Sepp Hochreiter, Andreas Mayr, and Günter Klambauer. Vn-egnn: E (3)-and se (3)-equivariant graph neural networks with virtual nodes enhance protein binding site identification. *Journal of Cheminformatics*, 2025.

Zachary Smith, Michael Strobel, Bodhi P Vani, and Pratyush Tiwary. Graph attention site prediction (grasp): Identifying druggable binding sites using graph neural networks with attention. *Journal of chemical information and modeling*, 64(7):2637–2644, 2024.

Javier S Utgés and Geoffrey J Barton. Comparative evaluation of methods for the prediction of protein–ligand binding sites. *Journal of Cheminformatics*, 16(1):126, 2024.

Fabien Vincent, Arsenio Nueda, Jonathan Lee, Monica Schenone, Marco Prunotto, and Mark Mercola. Phenotypic drug discovery: recent successes, lessons learned and new directions. *Nature Reviews Drug Discovery*, 21(12):899–914, 2022.

Jian Wang and Nikolay V. Dokholyan. Unified protein–small molecule graph neural networks for binding site prediction. *Proceedings of the National Academy of Sciences*, 123(10):e2524913123, 2026. doi: 10.1073/pnas.2524913123. URL https://www.pnas.org/doi/abs/10.1073/pnas.2524913123.

Tong Wang, Xinheng He, Mingyu Li, Yatao Li, Ran Bi, Yusong Wang, Chaoran Cheng, Xiangzhen Shen, Jiawei Meng, He Zhang, et al. Ab initio characterization of protein molecular dynamics with ai2bmd. *Nature*, 635(8040):1019–1027, 2024a.

Yusong Wang, Tong Wang, Shaoning Li, Xinheng He, Mingyu Li, Zun Wang, Nanning Zheng, Bin Shao, and Tie-Yan Liu. Enhancing geometric representations for molecules with equivariant vector-scalar interactive message passing. *Nature Communications*, 15(1):313, 2024b.

Shuo Zhang and Lei Xie. Protein language model-powered 3d ligand binding site prediction from protein sequence. *arXiv preprint arXiv:2312.03016*, 2023.

Zhijun Zhang, Lijun Quan, Junkai Wang, Liangchen Peng, Qiufeng Chen, Bei Zhang, Lexin Cao, Yelu Jiang, Geng Li, Liangpeng Nie, et al. Labind: Identifying protein binding ligand-aware sites via learning interactions between ligand and protein. *Nature Communications*, 16(1):7712, 2025.

A Related Work

Models of Binding Site Prediction. Protein binding site prediction has been studied through geometric pocket detection, learning-based surface scoring, and deep residue classification. Classical methods such as Fpocket and P2Rank identify ligandable regions from protein surface geometry or handcrafted local descriptors, and remain strong baselines due to their explicit use of surface locality (Le Guilloux et al., 2009; Kriválk & Hoksza, 2018). Recent deep models further improve accuracy by using voxelized structures (Jiménez et al., 2017; Kandel et al., 2021; Aggarwal et al., 2021a), protein language models (Fang et al., 2023; Carbery et al., 2024), or predicted contacts (Zhang & Xie, 2023). Most of them still formulate BSP as residue classification, surface-point scoring, or post-hoc clustering. Several recent works attempt to introduce entities to represent ligands: VN-EGNN (Sestak et al., 2025) uses virtual nodes to model latent binding pockets, PocketAnchor (Li et al., 2023) uses anchor probe points, and ligand-aware methods such as LABind (Zhang et al., 2025) and FABind (Pei et al., 2023) condition prediction on a given ligand.

Graph Neural Networks. Graph neural networks have become one of the most successful frameworks for molecular interaction modeling. In neural interatomic potentials and protein structure modeling, architectures such as SchNet (Schütt et al., 2017), DimeNet (Gasteiger et al., 2020), GemNet (Gasteiger et al., 2021; 2022), EGNN (Satorras et al., 2021), PaiNN (Schütt et al., 2021), NequIP (Batzner et al., 2022), MACE (Batatia et al., 2022), and ViSNet (Wang et al., 2024b) achieve strong performance by propagating messages over local geometric neighborhoods with distance (Schütt et al., 2017; Satorras et al., 2021), angular features (Gasteiger et al., 2020; 2021; 2022),

and equivariant representations (Schütt et al., 2021; Batzner et al., 2022; Batatia et al., 2022; Wang et al., 2024b; Satorras et al., 2021). Recently, AI2BMD (Wang et al., 2024a) has demonstrated the capability of applying machine-learned interatomic potentials to protein simulation. These results suggest that the strong expressive capacity of GNNs should also be applicable to BSP, a task that requires modeling protein–ligand interactions.

Object Detection. One-stage detectors formulate object detection as dense prediction: anchors or grid cells are first placed over images, after which each candidate directly predicts category scores and box displacements. Representative methods such as YOLO (Redmon et al., 2016), SSD (Liu et al., 2016), and RetinaNet (Lin et al., 2017) follow this efficient paradigm. Recent improvements often refine the training objective rather than the detection pipeline. For example, Generalized Focal Loss models the uncertainty in offset prediction (Li et al., 2020). Similarly, ProgLoss progressively rebalances loss contributions during training (Sapkota et al., 2025). In 3D detection, VoteNet (Qi et al., 2019) extends this "predict offsets from candidates" idea to point clouds by letting seed points vote toward object centers before generating 3D bounding boxes.

B LIMITATIONS

CQFiner uses local query neighborhoods, which may limit its coverage of large binding sites. Query coordinates represent pocket centers; the residue-mask head describes site membership but does not explicitly reconstruct the continuous three-dimensional pocket surface.

C EFFECT OF DISTANCE GATING IN RESIDUE CLASSIFICATION

As shown in Fig. 5, the main effect of the distance gate is to enhance the locality of CQFiner. Distance gating decreases the predicted binding-site probability of distant residues while increasing that of nearby residues. Since a query has only a limited receptive field, its predictions for distant residues are unreliable. The distance gate suppresses these predictions, thereby facilitating training.

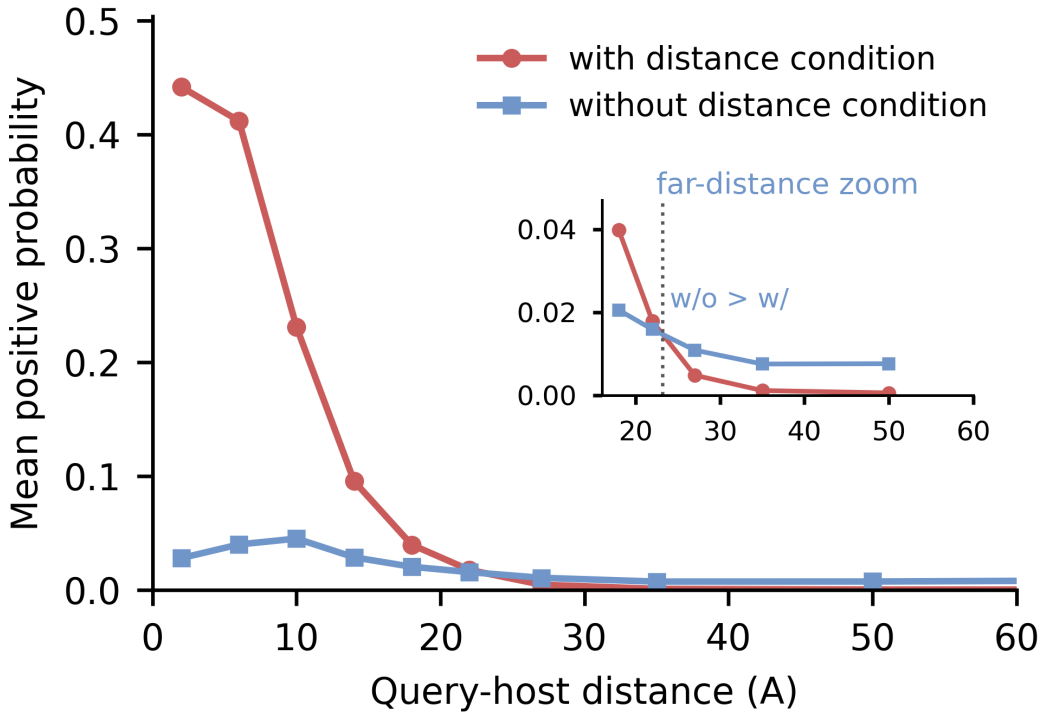

Figure 5: Effect of distance conditioning in the query–residue dot product.

D DIAGNOSTIC PROTOCOLS

Query features and loss allocation. Effective rank is the exponential of the entropy of the normalized singular values of the uncentered query-feature matrix. Cosine similarity averages all distinct query pairs within each protein. Fig. 1b uses the 32-query baseline at epoch 60 and resamples its original two-protein batches for confidence intervals. Fig. 4a compares 200-query CQFiner variants at their best validation-AP checkpoints (local: epoch 17; full: epoch 31), with paired protein bootstrap intervals. Both use 2,000 resamples. Loss-allocation plots retain each model's native loss reduction: Huber loss for the baseline and stage-0 coordinate MSE for CQFiner, without progressive objective weights. Distances are measured from initial queries to site centers; CQFiner's supervision cutoff instead uses the nearest ligand atom.

Memory measurements. Both figures use the same three proteins with 498–500 residues, selected near the sampled validation cohort's 90th percentile. Each condition runs in a fresh process on an RTX 4090 with batch size 1, FP16 autocast, and activation checkpointing disabled. Peak allocated memory is measured over a forward/backward/AdamW step after three warmup steps. Fig. 1d uses six-layer, 128-channel backbone allocation probes with bidirectional residue–query edges and local query–query edges; backward computation starts from scalar outputs rather than task losses. Fig. 4c measures the complete two-stage CQFiner models with directed residue-to-query edges, no query–query edges, and their full training objectives. The first diagnoses dense-graph scaling; the second quantifies memory savings within CQFiner. Medians and ranges use all three proteins; an OOM on any protein marks that configuration as OOM.

E PRIOR INJECTION AND TEST-TIME SCALING PROTOCOLS

Surface prior. To evaluate whether the models benefit from more informative query placement, we construct an oracle spatial prior using known binding-site locations. Let $\{\mathbf{p}_m\}_{m=1}^{M}$ be the valid surface points and $\{\mathbf{y}_s\}_{s=1}^{S}$ the ground-truth ligand centers, computed from ligand heavy atoms. We define the probability of sampling surface point m as

$$w_m = \frac{1}{S}\sum_{s=1}^{S}\frac{\exp\left(-\|\mathbf{p}_m-\mathbf{y}_s\|_2^2/(2\sigma^2)\right)}{\sum_{j=1}^{M}\exp(-\|\mathbf{p}_j-\mathbf{y}_s\|_2^2/(2\sigma^2))}, \tag{2}$$

with $\sigma = 6$ Å. Each site's Gaussian weights are normalized over the surface before averaging, giving each site equal weight in the prior. Site coordinates are used only to construct the sampling distribution; they are not passed to the model as features or used for ranking.

Fixed-budget prior injection. Both models are trained and evaluated with $N_q = 200$ queries. For prior fraction $\rho \in \{0, 0.25, 0.5, 0.75, 1\}$, we take ρN_q surface anchors sampled from w_m and $(1-\rho)N_q$ anchors sampled uniformly from the surface, both with replacement. Each anchor receives an outward normal offset sampled uniformly from 1.5–4 Å and a tangent perturbation with a random unit direction and normally distributed amplitude of standard deviation 0.5 Å. Normals are estimated by local PCA using up to 16 surface neighbors. These perturbations follow the training-time query sampler. For each protein and sampling seed, we generate background and prior query pools once and take the required prefix from each pool at every prior fraction. Both models receive identical initial coordinates. Their weights and post-processing settings remain fixed, using checkpoints selected by validation $AP_{0.5}$. Thus, the experiment changes query placement while keeping the query budget and trained models unchanged.

Test-time query scaling. The separate 32-query models in Fig. 3a are evaluated with 32, 64, 128, 200, 400, and 800 queries. Larger budgets use longer prefixes of the same background pool, preserving the queries already present at smaller budgets. Their validation-selected checkpoints use refiner feature normalization disabled, as during training; the 200-query prior-injection models retain the default normalization. These controlled experiments use random surface sampling with the perturbations above, whereas default benchmark inference uses farthest point sampling.

Paired evaluation. We fix 128 proteins from COACH420 and 128 from HOLO4K, sampled without replacement from the existing evaluation cohort with seed 20260926. Query-sampling seeds

42, 43, and 44 are paired across models and conditions. $AP_{0.5}$ is computed over each dataset's pooled predictions for each seed, then averaged equally across the two datasets and three seeds. Prior-injection gains are measured relative to each model's own $\rho = 0$ result. The 95% confidence intervals use 1,000 whole-protein bootstrap resamples within each dataset, preserving all model, condition, and sampling-seed pairings. The oracle prior provides a controlled test of how each model uses site-location information supplied through query placement.

F Implementation Details

Table 4: Default implementation settings for the residue ViSNet CQFiner model.

Component	Setting	Value
Residue graph	Node type	Residue C_α
Residue feature	Frozen PLM embedding	ESM, 1280 dimensions
Surface queries	Train / eval queries	200 / 200
Training offsets	Normal offset / tangent jitter	1.5∼4.0 Å / 0.5 Å
Graph cutoff	Residue–residue / residue–query	10 Å / 10 Å
Message direction	Residue-to-query / query-to-residue	On / Off
Backbone	GNN / layers / hidden dimension	ViSNet / 6 / 128
Refinement	GNN / layers / neighbor cap	EGNN / 3 / 32
Query ranking	Distance bins / bin size	49 / 0.5 Å
Offset/mask supervision	Initial ligand-atom distance	< 10 Å
Mask head	Projection dimension / RBF count	128 / 32
Mask distance scale	Range / initialization	2∼30 Å / 12 Å
Post-processing	Query NMS / mask NMS	6 Å / 7 Å
Post-processing	Score threshold / mask threshold	0.4 / 0.5
Optimization	Optimizer / learning rate	AdamW / 2×10^{-4}
Optimization	Betas / weight decay	$(0.9, 0.95)$ / 0
Optimization	LR reduction factor / patience	0.8 / 10 epochs
Training	Maximum epochs / GPUs	60 / 2
Training	Batch size per GPU / effective batch	8 / 16
Training	Precision / random seed	16-bit mixed precision / 42

This section summarizes the default residue-level ViSNet configuration used in the main experiments, including the three-layer EGNN refinement stage. Unless otherwise stated, both training and evaluation use 200 queries per protein. The 32-query training and modified evaluation sampling used in Fig. 3 are separate experimental settings.

Data construction. Each example is a protein with one or more annotated ligand-binding sites. We remove ligand hydrogens and use the heavy atoms belonging to each site to define the site center

$$\mathbf{y}_s = \frac{1}{|\mathcal{L}_s|} \sum_{\ell \in \mathcal{L}_s} \mathbf{r}_\ell, \tag{3}$$

where $\mathcal{L}_s$ is the set of heavy ligand atoms assigned to site s. In the default residue setting, each residue node is a standard residue represented by its C_α coordinate. Its frozen 1280-dimensional ESM feature is projected to the hidden dimension and combined with a learned residue-type embedding. Per-site residue masks are read from the dataset annotations and retained for mask supervision and AP evaluation.

The data object also stores solvent-accessible surface atom coordinates and their SASA values. These surface points are used to initialize queries; the default protein graph uses residue nodes. Graph inputs are cached after preprocessing, while interaction edges are constructed at runtime.

For LIGYSIS evaluation, the prepared data provide site-center proxy atoms instead of ligand heavy atoms; both DCC and DCA therefore measure distance to these supplied site centers.

Surface queries and runtime graph. For every protein, CQFiner samples N_q query nodes from valid surface points. During training, surface points are sampled randomly with replacement; validation and testing use farthest point sampling for deterministic coverage. A sampled point $\mathbf{p}$ is lifted

away from the protein surface by

$$\mathbf{q}_0 = \mathbf{p} + \alpha \mathbf{n} + \mathbf{e}_{\tan}, \tag{4}$$

where $\mathbf{n}$ is the outward surface normal estimated by local PCA using up to 16 neighboring surface points. During training, α is sampled uniformly between 1.5 and 4.0 Å. Tangent jitter uses a random unit tangent direction with a normally distributed scalar amplitude of standard deviation 0.5 Å. Evaluation uses the midpoint offset, 2.75 Å, without tangent jitter. Query nodes share a learned input embedding and are distinguished by their coordinates and local geometric neighborhoods.

The coarse graph contains residue–residue edges and directed residue-to-query edges within 10 Å, with a neighbor cap of 256. Query-to-residue and query-to-query edges are disabled. The fine-stage graph contains only residue-to-query edges. It is constructed at the coarsely displaced query positions using a 10 Å cutoff and at most 32 nearest residues per query. Its edge indices remain fixed across the three refinement layers, while edge distances and messages are recomputed from the updated coordinates.

Backbone and query refinement. The coarse stage uses a 6-layer ViSNet backbone with hidden dimension 128, 64 radial basis functions, 8 attention heads, and $l_{\max} = 2$. Residue features and the shared query embedding are projected into the hidden space and passed through the directed geometric interaction encoder. Query coordinates remain fixed within this stage. A displacement head aggregates residue-to-query edge contributions by their mean and predicts the coarse proposal

$$\mathbf{q}_q^{(0)} = \mathbf{q}_{0,q} + \Delta \mathbf{q}_q. \tag{5}$$

A three-layer EGNN then updates query features and coordinates using the coarse residue features as a fixed context. Residue coordinates and features are held unchanged throughout refinement. The refiner uses mean message aggregation, feature and coordinate normalization, and dropout 0.1. Its initial coordinate-update scale is 0.01. During training, the coarse proposal coordinates are detached before refinement; feature paths remain differentiable. The final site-center proposal is $\hat{\mathbf{y}}_q = \mathbf{q}_q^{(3)}$.

CQFiner ranks query proposals with a distance-distribution head. It predicts a categorical distribution over $B = 49$ distance bins with bin size $\delta = 0.5$ Å. The expected distance and ranking score are

$$\hat{d}_q = \sum_{b=0}^{B-1} p_{qb} b\delta, \qquad a_q = \exp\left(-\frac{\hat{d}_q}{(B-1)\delta}\right). \tag{6}$$

The final refinement-layer scores are used for NMS and evaluation.

Residue-mask head. Residue-level site masks are predicted with the distance-gated dot-product head. For query feature $\mathbf{h}_q$, residue feature $\mathbf{h}_i$, and final query–residue distance d_{qi}, the mask logit is

$$m_{qi} = \frac{1}{\sqrt{D}} \left[f_q(\mathbf{h}_q) \odot f_h(\mathbf{h}_i) \odot g(\phi(d_{qi}/r_q)) \right]^\top \mathbf{1}, \qquad D = 128. \tag{7}$$

Here f_q and f_h are two-layer MLPs, and $g(\cdot) = 2\sigma(\mathrm{MLP}(\cdot))$ is a vector-valued distance gate. The expansion ϕ contains 32 Gaussian radial basis functions over the normalized distance. A query-conditioned MLP predicts the scale r_q between 2 and 30 Å, initialized to 12 Å. Thus, 30 Å bounds the learned distance scale; it is not a hard cutoff that removes query–residue mask pairs. Applying a sigmoid to m_{qi} gives the residue probability. The head is class-agnostic and has no additional residue Transformer encoder.

Training objective. Each query is assigned to the nearest annotated site center at its initial position. Displacement and mask supervision are restricted to queries initially within 10 Å of any ligand heavy atom in the protein. This selection and the displacement target remain fixed during refinement. The ligand-atom criterion differs from distance to the assigned site center, which can exceed 10 Å.

For a prediction stage k, let $\mathcal{Q}_s$ contain the supervised queries assigned to site s, and let $\mathcal{S}_+$ contain sites with nonempty $\mathcal{Q}_s$. Displacement loss averages coordinate-wise squared errors within each site and then across the nonempty sites in the batch:

$$\mathcal{L}_{\mathrm{disp}}^{(k)} = \frac{1}{|\mathcal{S}_+|} \sum_{s \in \mathcal{S}_+} \frac{1}{|\mathcal{Q}_s|} \sum_{q \in \mathcal{Q}_s} \frac{\|\mathbf{q}_q^{(k)} - \mathbf{y}_s\|_2^2}{3}. \tag{8}$$

Stage $k = 0$ denotes the coarse proposal, and $k = 1, 2, 3$ denote successive EGNN updates.

Distance-distribution supervision covers all queries. At each prediction state, the distance to the nearest site center is clipped to $[0, 24]$ Å and interpolated between adjacent bins. This yields the soft target $\tilde{\mathbf{p}}_q^{(k)}$. The coarse distance head is supervised at the initial query coordinates, whereas refinement heads use their corresponding updated coordinates. Per-query cross-entropy is averaged within groups and then equally across groups:

$$\mathcal{L}_{\text{dist}}^{(k)} = -\frac{1}{|\mathcal{G}|} \sum_{G \in \mathcal{G}} \frac{1}{|G|} \sum_{q \in G} \sum_{b=0}^{B-1} \tilde{p}_{qb}^{(k)} \log p_{qb}^{(k)}. \tag{9}$$

The nonempty groups $\mathcal{G}$ comprise each initially supervised site group and one background group per protein for queries outside the supervision cutoff. The initial grouping is retained across stages; background queries still receive distance-distribution targets.

Both displacement and distance-distribution losses combine the coarse loss with the mean of all three refinement-layer losses:

$$\overline{\mathcal{L}}_u = \frac{1}{2} \left(\mathcal{L}_u^{(0)} + \frac{1}{3} \sum_{k=1}^{3} \mathcal{L}_u^{(k)} \right), \qquad u \in \{\text{disp}, \text{dist}\}. \tag{10}$$

Mask supervision uses the final refined query features and positions. Each supervised query with an available target mask is paired with the residue mask of its initially assigned site. The mask objective is binary cross-entropy plus Dice loss, averaged over the supervised queries:

$$\mathcal{L}_{\text{site}} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{Dice}}. \tag{11}$$

The complete objective uses progressive weighting:

$$\mathcal{L} = \mathcal{L}_{\text{site}} + (1 - \tau)\overline{\mathcal{L}}_{\text{dist}} + \tau\overline{\mathcal{L}}_{\text{disp}}, \tag{12}$$

where τ is the completed fraction of the planned training steps. Query contrastive and query-affinity losses are disabled in the default configuration.

Optimization and checkpoint selection. We train for up to 60 epochs using AdamW with learning rate 2×10^{-4}, betas $(0.9, 0.95)$, and zero weight decay. ReduceLROnPlateau monitors validation loss, with factor 0.8, patience 10 epochs, and minimum learning rate 10^{-6}. Training uses two GPUs, eight proteins per GPU, no gradient accumulation, and 16-bit mixed precision. The effective batch size is 16 proteins, and the random seed is 42. Reported default-model and ablation results use checkpoints selected by the highest UniSite validation $\text{AP}_{0.5}$.

Inference and post-processing. Default inference samples 200 surface queries with farthest point sampling. One ViSNet pass produces the coarse features and displacements, followed by the three-layer EGNN refiner using the retained residue features. No second ViSNet pass is used by the default model. Final query centers are ranked by a_q and filtered by NMS with a 6 Å radius for query-center DCC/DCA.

Mask prediction uses final queries with scores of at least 0.4. Their masks are generated before grouping query centers with a separate 7 Å NMS radius. Within each group, residue probabilities are fused by a score-weighted average and thresholded at 0.5. The highest query score gives the group score, and groups with empty binary masks are discarded. These grouped masks are used for AP evaluation; the 6 Å query-center evaluation branch is separate.

For the separate site-center evaluation, residue probabilities are aggregated by the maximum query-score-weighted probability. Residues above probability 0.3 are clustered using average linkage with a 15 Å distance threshold. Site centers use the volume-weighted centroid of a tetrahedralization of the convex hull, and the residue-coordinate mean when a valid hull cannot be computed. These mask-derived centers are distinct from the direct query centers used for the main DCC/DCA results.