ASYMMETRY-AWARE REPRESENTATION LEARNING FOR TCR-EPITOPE BINDING PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

T cell receptors (TCRs) play a vital role in immune recognition by binding specific epitopes. Accurate prediction of TCR–epitope binding specificity is fundamental for advancing immunology research. Existing methods commonly formulate TCR–epitope interaction prediction as a dual-encoder matching problem, in which TCRs and epitopes are independently encoded and subsequently aligned in a shared representation space. However, this formulation overlooks the representational asymmetry between structurally complex TCRs and lower-complexity epitopes, introducing unnecessary modeling overhead and hindering effective alignment. To address this issue, we propose an asymmetry-aware representation learning method that combines structured TCR encoding with discrete epitope modeling, reducing joint modeling complexity and recasting heterogeneous alignment as a discriminative mapping to the candidate epitope space. Specifically, the model uses a frozen protein language model to extract contextualized CDR3 representations from full-length α- and β-chain sequences. These representations are then encoded by weight-shared multiscale convolutions and independent global branches to capture cross-chain local patterns and chain-specific context, respectively. Candidate epitopes are represented as discrete categories in a finite prediction space. The model is optimized with complementary supervision to learn the global TCR-to-epitope mapping while refining the relative distinction between target and competing epitopes. Comparative experiments against multiple baseline methods on several benchmark datasets demonstrate the effectiveness of the proposed method.

1 INTRODUCTION

T-cell receptors (TCRs) are central to adaptive immunity by recognizing antigenic peptides presented by major histocompatibility complex (MHC) molecules on antigen-presenting cells. Accurate prediction of TCR–epitope binding specificity is therefore important for understanding immune recognition mechanisms and has broad applications in immune repertoire analysis, antigen discovery, vaccine development, and antigen-specific immunotherapy Weber et al. (2024); Qi et al. (2025); Lu et al. (2026). As illustrated in Fig. 1(a), the TCR–epitope recognition process determines whether a T cell can respond to a specific antigen and serves as a foundation for antitumor immune responses. Approximately 95% of TCRs are composed of one α-chain and one β-chain, each containing three complementarity-determining regions (CDRs), in which CDR1 and CDR2 are conserved and CDR3 is the primary region of antigen contact Croce et al. (2024); Lu et al. (2026). The CDR3 region are generated through somatic V(D)J recombination producing substantial TCR diversity and determining T cell function. Consequently, TCR–epitope interactions are jointly determined by the CDR3 regions of the paired α- and β-chains, as illustrated in Fig. 1(b)–(c) Croce et al. (2024); Kim et al. (2024). Therefore, a given TCR can be described by its paired CDR3 α- and CDR3 β regions, while the recognized antigen is represented by the corresponding epitope sequence.

Based on these sequence representations, prediction of TCR–epitope binding aims to determine whether a given TCR can recognize a particular epitope, as illustrated in Fig. 1(d). In practice, the task can be viewed as learning the relationship between TCR sequence patterns and epitope identity or sequence, and assigning a recognition score to each TCR–epitope pair. Such computational prediction provides a high-throughput, low-cost and scalable complement to experimental assays, whose large-scale application remains constrained high experimental cost and low-throughput.

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.

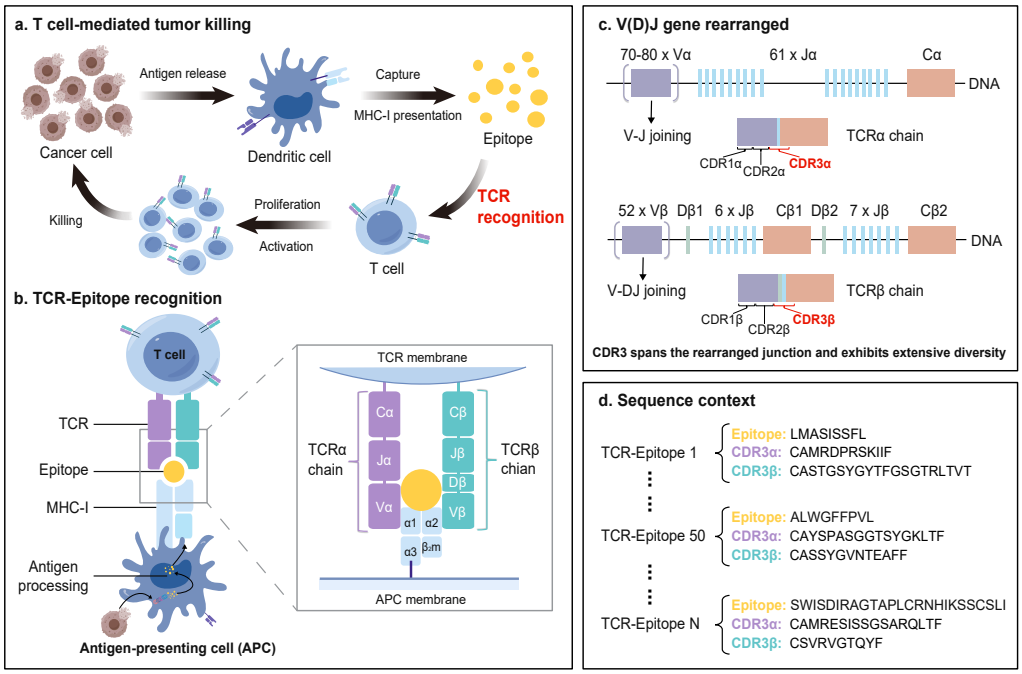


Figure 1: Overview of TCR–epitope binding prediction and sequence representation. (a) T-cell-mediated immune response initiated by antigen recognition. (b) Recognition of an epitope–MHC complex by the paired TCR α- and β-chains. (c) Generation of diverse TCR sequences through V(J)/V(D)J recombination, with CDR3 regions forming the most variable parts of the receptor. (d) Representative TCR–epitope pairs described by epitope, CDR3α, and CDR3β sequences.

Existing methods commonly formulate TCR–epitope prediction as a pairwise representation learning problem, in which TCR and epitope sequences are encoded into learned representations and subsequently matched to estimate recognition specificity. Recent methods employ convolutional architectures, attention mechanisms, Transformers, and pretrained protein language models to capture sequence context and TCR–epitope binding prediction Jokinen et al. (2023); Jiang et al. (2024); Wu et al. (2024); Kim et al. (2024); Li et al. (2025); Fu et al. (2025). These approaches have substantially advanced sequence representation learning and enabled increasingly sophisticated modeling of molecular interaction patterns. A more comprehensive review of existing TCR–epitope binding prediction methods, including their modeling strategies and methodological characteristics, is provided in Appendix A. A common design pattern, however, is to allocate substantial learnable capacity to both the TCR and epitope sides and subsequently align the resulting heterogeneous representations.

This formulation does not fully exploit an important asymmetry of the specificity-classification setting. TCRs exhibit enormous sequence diversity and encode multilevel information across paired α- and β-chains, making it necessary to capture both locally shared recognition motifs and chain-specific sequence context Croce et al. (2024); Kim et al. (2024); Wu et al. (2024); Lu et al. (2026). In contrast, under finite-candidate classification settings, epitopes constitute a substantially smaller and more discrete candidate space and primarily provide the target identities against which heterogeneous TCR representations must be discriminated. Treating both sides with similarly elaborate representation-learning pipelines may therefore introduce unnecessary modeling complexity while increasing the burden of aligning two structurally different representation spaces. More importantly, recent systematic benchmarks show that increasingly sophisticated sequence encoders do not by themselves ensure robust generalization: performance remains strongly affected by training-data composition, epitope frequency, negative-sample construction, and whether evaluation involves unseen epitopes or genuinely independent data Deng et al. (2023); Nielsen et al. (2024); Feng et al. (2025); Drost et al. (2025); Lu et al. (2026).

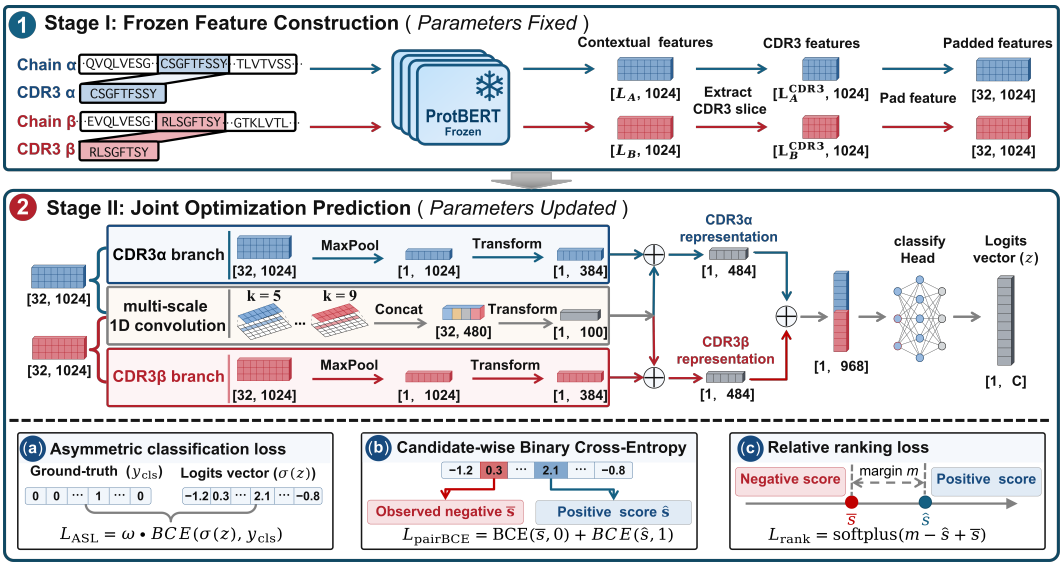


Figure 2: Overview of the proposed AsymTCR. A frozen protein language model first constructs contextualized CDR3 features, followed by chain-specific global encoding and weight-shared multiscale local encoding. The resulting TCR representation is mapped to the candidate epitope space and optimized with asymmertric classification, pairwise, and ranking objectives.

To address this issue, we propose **AsymTCR**, an **asymmetry-aware representation learning** method that explicitly differentiates the modeling roles of TCRs and epitopes. Instead of learning two equally complex representation spaces and aligning them afterward, AsymTCR performs structured representation learning on the TCR while adopting lightweight discrete modeling on the epitope. Specifically, a frozen protein language model is used to extract contextualized CDR3 representations from full-length α- and β-chain sequences, and these representations are further encoded by weight-shared multiscale convolutions and independent global branches to capture cross-chain local patterns and chain-specific context, respectively. Candidate epitopes are modeled in a lightweight discrete space, transforming heterogeneous bilateral matching into a direct discriminative mapping from structured TCR representations to candidate epitopes. AsymTCR is further optimized with complementary supervision to learn the global TCR-to-epitope mapping while refining the relative distinction between target and competing epitopes. Extensive experiments on multiple benchmark datasets and comparisons with diverse baseline methods demonstrate the effectiveness of AsymTCR.

The main contributions of this work are summarized as follows:

- We revisit TCR–epitope binding prediction from the perspective of *representational asymmetry*, challenging the prevailing symmetric modeling paradigm.
- We propose an asymmetry-aware method that combines rich structured TCR encoding with lightweight epitope modeling, thereby reducing bilateral modeling complexity and recasting heterogeneous alignment as a direct discriminative mapping.
- We develop a structured TCR encoder that integrates contextualized full-chain representations, shared multiscale local modeling, and chain-specific global information, together with complementary supervision for improved target association and inter-candidate discrimination.

2 Method

2.1 Problem Formulation

TCR–epitope binding prediction aims to determine whether a given TCR specifically recognizes a particular epitope. Given a TCR T_i and an epitope E_j, the task is formulated as $s_{ij} = f(T_i, E_j)$,

where $s_{ij} \in [0,1]$ denotes the predicted specificity score, with a larger value indicating a higher likelihood of specific recognition. Let $\Sigma = \{\mathrm{A}, \mathrm{C}, \mathrm{D}, \mathrm{E}, \mathrm{F}, \mathrm{G}, \mathrm{H}, \mathrm{I}, \mathrm{K}, \mathrm{L}, \mathrm{M}, \mathrm{N}, \mathrm{P}, \mathrm{Q}, \mathrm{R}, \mathrm{S}, \mathrm{T}, \mathrm{V}, \mathrm{W}, \mathrm{Y}\}$ denote the standard amino-acid alphabet. Each TCR consists of paired α- and β-chains and is represented by four sequence inputs,

$$T_i = \left(A_i^L, A_i^C, B_i^L, B_i^C\right), \tag{1}$$

where A_i^L and B_i^L denote the long-form α- and β-chain sequences, while A_i^C and B_i^C denote their corresponding CDR3 regions, with $A_i^C \subset A_i^L$ and $B_i^C \subset B_i^L$. The long-form chains provide broader sequence context, whereas the CDR3 regions capture highly variable sequence patterns closely associated with antigen recognition.

The epitope $E_j \in \Sigma^{L_j^E}$ is represented as a short antigenic peptide sequence. In the finite-candidate setting considered in this work, all epitopes are drawn from $\mathcal{E} = \{E_1, E_2, \ldots, E_C\}$, where C denotes the number of candidate epitopes. Accordingly, a prediction instance is defined as $x_{ij} = \left(A_i^L, A_i^C, B_i^L, B_i^C, E_j\right)$, with corresponding label y_{ij}, and the training set is $\mathcal{D} = \left\{\left(A_i^L, A_i^C, B_i^L, B_i^C, E_j, y_{ij}\right)\right\}_{i=1}^{n}$. For each TCR, the model jointly estimates

$$\mathbf{s}_i = [s_{i1}, s_{i2}, \ldots, s_{iC}]^\top, \qquad \hat{j} = \arg\max_j s_{ij}. \tag{2}$$

Although TCRs and epitopes are both amino-acid sequences, they have different structural characteristics and roles in this prediction setting. A TCR contains multiple coupled sequence components and requires structured representation learning, whereas epitopes belong to a finite candidate space. This asymmetry motivates our subsequent design, which focuses representation learning on the TCR side and models candidate epitopes through the prediction space.

2.2 Asymmetry-Aware TCR Representation and Candidate Prediction

As illustrated in Fig. 2, AsymTCR comprises two stages. Stage I uses frozen ProtBERT to extract and pad contextualized CDR3 features from the long-form α- and β-chains. Stage II combines chain-specific global encoding with weight-shared multiscale local encoding, fuses the two chain representations, and maps the resulting paired-TCR representation to candidate-epitope logits. The trainable modules are jointly optimized using asymmetric classification, candidate-wise binary cross-entropy, and relative ranking losses.

2.2.1 Stage I: Contextualized CDR3 Feature Construction

As shown in the Stage I block of Fig. 2, the long-form TCR chains are first encoded by a frozen protein language model, after which the CDR3-aligned contextual features are extracted and padded to a common length. This construction preserves the surrounding sequence context of each CDR3 residue instead of encoding the isolated CDR3 sequence alone.

Let $\chi \in \{\alpha, \beta\}$ denote the chain type, with $S_i^\alpha = A_i^L$ and $S_i^\beta = B_i^L$, where $S_i^\chi \in \Sigma^{L_i^\chi}$ and L_i^χ is the length of chain χ. The frozen ProtBERT module shown on the left of Fig. 2 produces the contextual feature map

$$\mathbf{H}_i^\chi = F_{\mathrm{PLM}}(S_i^\chi) \in \mathbb{R}^{L_i^\chi \times d_p}, \qquad \chi \in \{\alpha, \beta\}, \tag{3}$$

where F_{PLM} denotes the pretrained protein language model and d_p is its hidden dimension.

Let $\mathcal{I}_i^\chi$ denote the residue-index set of the CDR3 region within S_i^χ. Corresponding to the "Extract CDR3 slice" and "Pad feature" operations in Fig. 2, the contextualized CDR3 feature is constructed as

$$\widetilde{\mathbf{X}}_i^\chi = \mathcal{P}_{L_0}\left(\mathbf{H}_i^\chi[\mathcal{I}_i^\chi, :]\right) \in \mathbb{R}^{L_0 \times d_p}, \tag{4}$$

where $\mathcal{P}_{L_0}(\cdot)$ denotes padding to length L_0. The padded feature matrices shown at the right end of Stage I therefore correspond to $\widetilde{\mathbf{X}}_i^\alpha$ and $\widetilde{\mathbf{X}}_i^\beta$, which are passed directly to Stage II. All parameters of F_{PLM} remain fixed during task-specific training.

2.2.2 Stage II: Local–Global TCR Representation Learning

As illustrated in Stage II of Fig. 2, each contextualized CDR3 feature is decomposed into chain-specific global context and cross-chain shared local patterns. The global branches use independent

transformations for the α- and β-chains, whereas the multiscale convolutional branch shares its parameters across chains.

For each chain $\chi \in \{\alpha, \beta\}$, global contextual information is obtained by sequence-wise max pooling followed by a chain-specific transformation:

$$\mathbf{g}_i^{\chi} = T_{\chi}\left(\mathrm{MaxPool}(\widetilde{\mathbf{X}}_i^{\chi})\right) \in \mathbb{R}^{d_g}, \tag{5}$$

where the independently parameterized T_α and T_β preserve chain-specific global information.

To capture local recognition patterns at multiple sequence scales, let $\mathcal{K} = \{k_r\}_{r=1}^{R}$ denote the convolutional kernel sizes and q_r the corresponding output-channel numbers. At each scale, the same convolutional parameters are applied to both chains:

$$\mathbf{U}_{i,r}^{\chi} = \phi\left(\mathrm{Conv}_{k_r}\left(\widetilde{\mathbf{X}}_i^{\chi}; \mathbf{\Theta}_r\right)\right) \in \mathbb{R}^{L_0 \times q_r}, \qquad r = 1, \ldots, R, \tag{6}$$

where $\mathbf{\Theta}_r$ is shared across the two chains and same padding preserves the sequence length. The scale-specific feature maps are concatenated as

$$\mathbf{M}_i^{\chi} = \mathrm{Concat}\left(\mathbf{U}_{i,1}^{\chi}, \ldots, \mathbf{U}_{i,R}^{\chi}\right) \in \mathbb{R}^{L_0 \times q}, \qquad q = \sum_{r=1}^{R} q_r, \tag{7}$$

and converted into a fixed-dimensional local representation through a shared pooling–projection transformation:

$$\mathbf{l}_i^{\chi} = T_{\mathrm{loc}}(\mathbf{M}_i^{\chi}) \in \mathbb{R}^{d_l}. \tag{8}$$

The global and local representations are first fused within each chain:

$$\mathbf{h}_i^{\chi} = [\mathbf{g}_i^{\chi} \| \mathbf{l}_i^{\chi}] \in \mathbb{R}^{d_g + d_l}, \qquad \chi \in \{\alpha, \beta\}, \tag{9}$$

and the two chain representations are then concatenated to form the paired-TCR representation:

$$\mathbf{h}_i = \left[\mathbf{h}_i^{\alpha} \| \mathbf{h}_i^{\beta}\right] \in \mathbb{R}^{2(d_g + d_l)}. \tag{10}$$

Finally, the classification head directly maps $\mathbf{h}_i$ to the finite candidate-epitope space:

$$\mathbf{z}_i = F_{\mathrm{cls}}(\mathbf{h}_i) = [z_{i1}, \ldots, z_{iC}]^{\top} \in \mathbb{R}^{C}, \tag{11}$$

followed by candidate-wise specificity scores

$$\mathbf{s}_i = \sigma(\mathbf{z}_i) = [s_{i1}, \ldots, s_{iC}]^{\top}, \qquad s_{ij} = \sigma(z_{ij}) = f(T_i, E_j). \tag{12}$$

This direct projection avoids an additional continuous epitope encoder and produces the shared logits and scores used by the optimization objectives described next.

2.2.3 Model Optimization

The three loss modules in Fig. 2(a)–(c) progressively supervise the same candidate-score space. The asymmetric classification loss first organizes all candidate epitopes, the candidate-wise pair loss anchors this global structure using an explicitly observed positive–negative pair, and the ranking loss further enforces their relative separation.

For a mini-batch of B samples, let j_i^{+} and j_i^{-} denote the observed positive and negative candidate indices for sample i, respectively, and define

$$t_{ij} = \mathbb{I}[j = j_i^{+}], \qquad z_i^{+} = z_{i,j_i^{+}}, \qquad z_i^{-} = z_{i,j_i^{-}}. \tag{13}$$

The asymmetric classification objective visualized in Fig. 2(a) learns global discrimination over all C candidate epitopes:

$$\mathcal{L}_{\mathrm{ASL}} = -\frac{1}{BC} \sum_{i=1}^{B} \sum_{j=1}^{C} \left[w_j t_{ij} \log s_{ij} + (1 - t_{ij}) s_{ij}^{\gamma} \log(1 - s_{ij}) \right], \tag{14}$$

where

$$w_j = \frac{N_+}{C \max(n_j, 1)}, \qquad N_+ = \sum_{j=1}^{C} n_j. \tag{15}$$

Here, n_j is the number of positive training samples for candidate E_j, w_j compensates for candidate-frequency imbalance, and γ down-weights easy negatives. Unselected candidates are treated as negatives only for closed-set optimization and are not assumed to be experimentally verified non-interactions.

The candidate-wise binary cross-entropy module in Fig. 2(b) provides direct supervision for the explicitly observed positive–negative pair:

$$\mathcal{L}_{\text{pair}} = -\frac{1}{2B} \sum_{i=1}^{B} \left[\log \sigma(z_i^+) + \log \left(1 - \sigma(z_i^-)\right) \right]. \tag{16}$$

This objective promotes the observed positive candidate while suppressing the observed negative candidate, thereby anchoring the global candidate structure with reliable pair-level supervision.

As visualized in Fig. 2(c), the relative ranking loss further requires the positive candidate to exceed the negative one by a margin:

$$\mathcal{L}_{\text{rank}} = \frac{1}{B} \sum_{i=1}^{B} \text{softplus} \left(m - z_i^+ + z_i^- \right), \tag{17}$$

where m is the ranking margin. Thus, $\mathcal{L}_{\text{pair}}$ calibrates the two observed candidates individually, while $\mathcal{L}_{\text{rank}}$ explicitly controls their relative ordering.

The three objectives shown in Fig. 2(a)–(c) are jointly optimized as

$$\mathcal{L} = \mathcal{L}_{\text{ASL}} + \lambda_{\text{pair}} \mathcal{L}_{\text{pair}} + \lambda_{\text{rank}} \mathcal{L}_{\text{rank}}. \tag{18}$$

Accordingly, the supervision progresses from organizing the complete candidate space, to calibrating observed candidate pairs, and finally to strengthening their relative ordering.

Consistent with the "Parameters Fixed" and "Parameters Updated" labels in Fig. 2, Stage I remains frozen, whereas the local–global representation modules in Stage II and the classification head are jointly optimized:

$$\mathbf{\Omega}^* = \arg\min_{\mathbf{\Omega}} \mathcal{L}. \tag{19}$$

At inference, the classification head shown at the right end of Fig. 2 produces all candidate scores in a single forward pass, and the predicted epitope is

$$\hat{j} = \arg \max_{j \in \{1,\ldots,C\}} s_{ij}. \tag{20}$$

All architectural and objective-specific hyperparameters are reported in Appendix B.

3 Experiments

3.1 Experimental Setup

We evaluated AsymTCR on four public TCR–epitope specificity datasets: the CDR3β+others setting of NM-Benchmark Lu et al. (2026), GSE332840 Ghoreyshi et al. (2026), IMMREP22 Meysman et al. (2023), and IMMREP23 Nielsen et al. (2024). NM-Benchmark was used as the primary benchmark because it integrates data from 21 public databases and studies and provides predefined cross-validation and independent test splits. Its CDR3β+others setting contains 249 epitopes and 5,294 binding TCRs before partitioning Lu et al. (2026). The remaining datasets were included to assess robustness across different data-generation protocols and antigen distributions. All datasets were processed using a consistent sequence-filtering and input-construction procedure. Further details are provided in Appendix C.

Performance was evaluated using three complementary metrics: area under the precision–recall curve (AUPRC), area under the receiver operating characteristic curve (AUROC), and accuracy

(ACC). AUPRC emphasizes the precision–recall trade-off under class imbalance, AUROC measures threshold-independent discriminative ability, and ACC reflects overall classification correctness. Detailed definitions and calculations are provided in Appendix D.

Under these unified evaluation settings, we compared AsymTCR with 16 configurations from 13 published methods: three TCRconv variants Jokinen et al. (2023), TCR-Rational Li et al. (2025), epiTCR Pham et al. (2023), VitTCR Jiang et al. (2024), two vibtcr variants Grazioli et al. (2023), TPepRet Wang et al. (2025), MINT Ullanat et al. (2026), TEINet Jiang et al. (2023), DAISY Yuan et al. (2026), TPBTE Wu et al. (2023), tcrLM Yu et al. (2026), TCR-BERT Wu et al. (2024), and TCR-H T. et al. (2024). These baselines cover convolutional, attention/Transformer, variational, protein-language-model, and physicochemical representation strategies. Baseline details are reported in Appendix E.

3.2 Main Results

Table 1: Performance comparison on NM-Benchmark under within-cohort cross-validation and independent testing. Values are mean ± standard deviation. Bold and underline indicate the best and second-best mean in each column.

	Within-cohort CV			Independent test		
Method	AUPRC	AUROC	ACC	AUPRC	AUROC	ACC
AsymTCR	**0.810±0.012**	**0.790±0.013**	**0.657±0.009**	**0.743±0.010**	**0.700±0.011**	<u>0.566±0.005</u>
TCRconv-fullAB	<u>0.787±0.007</u>	<u>0.738±0.008</u>	0.601±0.009	<u>0.732±0.004</u>	0.570±0.004	0.460±0.005
TCRconv-fullA	0.732±0.007	0.681±0.011	0.585±0.004	0.689±0.008	0.534±0.006	0.415±0.007
TCRconv-fullB	0.728±0.014	0.684±0.015	0.565±0.004	0.689±0.008	0.534±0.006	0.415±0.007
epiTCR	0.697±0.006	0.679±0.004	<u>0.622±0.007</u>	0.631±0.024	<u>0.599±0.030</u>	0.561±0.033
vibtcr-AB	0.500±0.003	0.502±0.005	0.500±0.000	0.492±0.007	0.497±0.009	0.508±0.011
VitTCR	0.579±0.011	0.588±0.012	0.561±0.010	0.548±0.028	0.537±0.019	0.531±0.012
TCR-BERT	0.669±0.009	0.666±0.010	0.617±0.009	0.637±0.019	0.597±0.036	**0.580±0.024**
TPepRet	0.633±0.013	0.638±0.013	0.594±0.010	0.553±0.027	0.546±0.025	0.538±0.020
vibtcr	0.503±0.004	0.500±0.002	0.500±0.001	0.485±0.026	0.484±0.034	0.489±0.027
TCR-Rational	0.583±0.025	0.590±0.026	0.564±0.016	0.570±0.013	0.563±0.013	0.541±0.015
TEINet	0.517±0.035	0.521±0.040	0.512±0.028	0.494±0.004	0.492±0.024	0.502±0.005
TCR-H	0.607±0.014	0.625±0.015	0.590±0.011	0.536±0.037	0.536±0.036	0.521±0.028
DAISY	0.555±0.011	0.560±0.011	0.537±0.010	0.546±0.021	0.551±0.020	0.539±0.015
MINT	0.520±0.014	0.528±0.014	0.500±0.000	0.491±0.009	0.484±0.011	0.500±0.000
TPBTE	0.500±0.001	0.501±0.002	0.500±0.000	0.506±0.007	0.508±0.012	0.500±0.000
tcrLM	0.509±0.005	0.500±0.001	0.492±0.004	0.508±0.000	0.500±0.000	0.492±0.000

Table 1 shows that AsymTCR achieves the best within-cohort performance on NM-Benchmark, with an AUPRC of 0.810, AUROC of 0.790, and ACC of 0.657, exceeding the second-best results by 2.3, 5.2, and 3.5 percentage points, respectively. On the independent test set, AsymTCR also achieves the highest AUPRC (0.743) and AUROC (0.700), while its ACC (0.566) is slightly lower than that of TCR-BERT (0.580). These results indicate a particularly clear advantage in threshold-independent discrimination on independently held-out data. Results on the other three datasets, GSE332840, IMMREP22, and IMMREP23, are reported in Appendix F. To summarize cross-dataset performance, we report the average results over all four datasets on Fig. 3 . From the figure, we can found that AsymTCR outperforms nearly all baselines in average AUPRC and all compared methods in average AUROC and ACC, with particularly large gains over several existing approaches. The improvement is most consistent for AUROC, where AsymTCR exceeds every baseline, suggesting that the proposed representation learning strategy provides robust discriminative ability across heterogeneous datasets rather than being specific to NM-Benchmark.

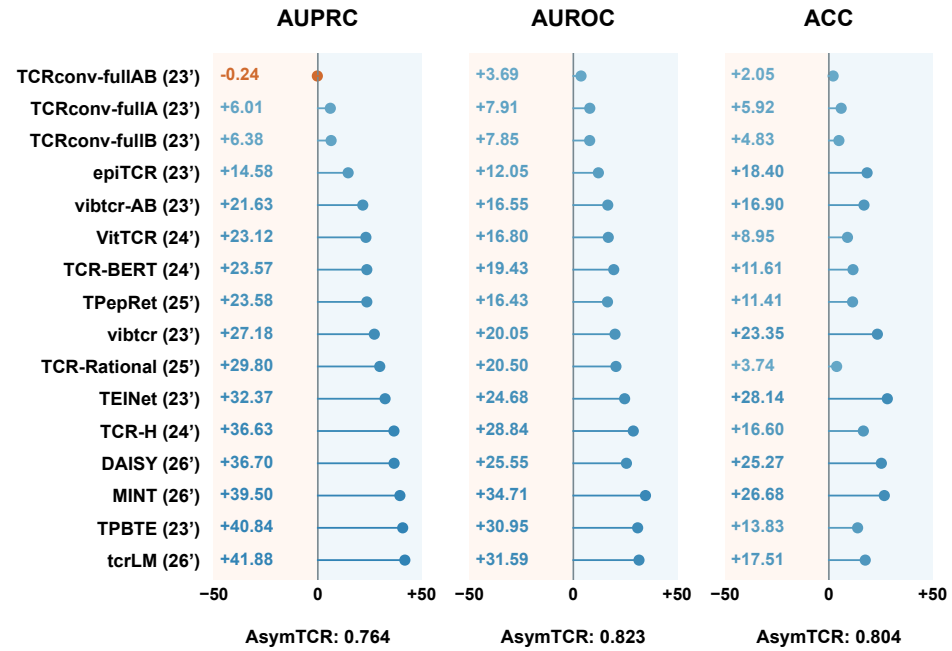
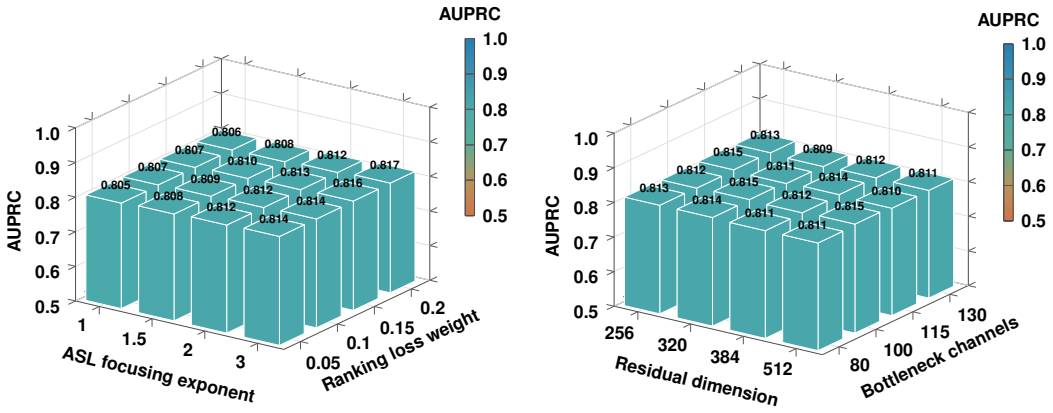


Figure 3: Mean percentage-point differences between AsymTCR and each baseline across four equally weighted datasets. Positive values favor AsymTCR, whose mean scores are shown below each panel.

3.3 Parameter Sensitivity



(a) Loss parameters

(b) Architecture parameters

Figure 4: **Parameter sensitivity on the NM-Benchmark.** Both panels use pooled five-fold out-of-fold AUPRC. The left panel varies the asymmetric-loss focusing exponent and ranking-loss weight; the right varies residual dimension and bottleneck channels. Each bar is labeled with its measured AUPRC.

Figure 4 examines the sensitivity of AsymTCR to representative loss and architecture hyperparameters on the NM-Benchmark. As shown in Fig. 4(a), AUPRC remains within a narrow range of 0.805–0.817 across different combinations of the ASL focusing exponent and ranking-loss weight. Increasing the ranking contribution generally yields modest improvements, whereas varying the focusing exponent produces only limited changes, indicating that performance is not dominated by a particular loss configuration. Similarly, Fig. 4(b) shows that varying the residual dimension and

bottleneck channels results in AUPRC values concentrated around 0.81, with no consistent benefit from simply increasing model capacity. Overall, the relatively flat response surfaces suggest that AsymTCR is stable within the examined parameter ranges rather than relying on a narrowly tuned configuration. Sensitivity results on the other datasets are provided in Appendix G.

3.4 Ablation Studies

Figure 5 evaluates the contribution of the main components in AsymTCR. Removing both CDR3α and CDR3β causes by far the largest performance drop, confirming that CDR3 sequences provide the primary recognition-related information. Single-chain ablations generally reduce AUPRC, indicating complementary rather than interchangeable contributions from the α- and β-chains. Removing long-form chain context while retaining CDR3 also degrades performance in most settings, supporting the benefit of contextualized CDR3 representation. Finally, replacing multiscale convolution with a single scale consistently lowers AUPRC, while removing the convolutional branch particularly affects the independent NM-Benchmark test, demonstrating the value of multiscale local pattern modeling. Ablation results under the other two evaluation metrics, AUROC and ACC, are provided in Appendix H.

	GSE 332840	IMMREP 22	IMMREP 23	NM-Benchmark training	NM-Benchmark independent
Full model (AsymTCR)	0.945 reference	0.709 reference	0.685 reference	0.812 reference	0.764 reference
Without α-chain (LongA + CDR3α)	0.882 (-6.28 pp)	0.641 (-6.82 pp)	0.610 (-7.42 pp)	0.738 (-7.38 pp)	0.746 (-1.80 pp)
Without β-chain (LongB + CDR3β)	0.959 (+1.40 pp)	0.600 (-10.83 pp)	0.593 (-9.15 pp)	0.741 (-7.09 pp)	0.652 (-11.22 pp)
Without LongA (retain CDR3α)	0.906 (-3.93 pp)	0.691 (-1.80 pp)	0.662 (-2.25 pp)	0.773 (-3.86 pp)	0.777 (+1.34 pp)
Without LongB (retain CDR3β)	0.873 (-7.19 pp)	0.681 (-2.81 pp)	0.651 (-3.35 pp)	0.779 (-3.32 pp)	0.732 (-3.21 pp)
Without LongA/B (retain CDR3α/β)	0.790 (-15.44 pp)	0.662 (-4.70 pp)	0.637 (-4.76 pp)	0.727 (-8.47 pp)	0.748 (-1.64 pp)
Without CDR3α/β	0.528 (-41.68 pp)	0.129 (-57.98 pp)	0.145 (-53.97 pp)	0.500 (-31.21 pp)	0.500 (-26.37 pp)
Without convolutional branch	0.929 (-1.56 pp)	0.671 (-3.81 pp)	0.653 (-3.12 pp)	0.822 (+0.98 pp)	0.726 (-3.84 pp)
Without context projection	0.940 (-0.49 pp)	0.708 (-0.12 pp)	0.682 (-0.28 pp)	0.811 (-0.07 pp)	0.773 (+0.85 pp)
Single-scale convolution (k = 5)	0.937 (-0.79 pp)	0.690 (-1.85 pp)	0.676 (-0.86 pp)	0.803 (-0.92 pp)	0.751 (-1.31 pp)

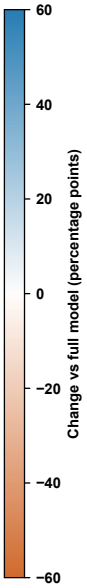


Figure 5: Ablation study of AsymTCR. Each row removes or simplifies one input component or representation module from the full model, and each cell reports the resulting AUPRC together with its percentage-point change relative to the full model.

4 Conclusion

In this work, we proposed AsymTCR, an asymmetry-aware framework for TCR–epitope binding prediction that combines structured paired-chain TCR representation learning with discrete candidate-epitope modeling. AsymTCR integrates contextualized CDR3 features, chain-specific global information, and shared multiscale local patterns, and directly maps the resulting TCR representation to the candidate epitope space. Experiments on multiple benchmark datasets demonstrate competitive and generally improved performance, while ablation studies support the contributions of paired-chain information, long-chain context, and multiscale local modeling. These results highlight the value of explicitly modeling TCR–epitope representational asymmetry. Extending the framework to unseen-epitope and open-set prediction remains an important direction for future work.

USE OF GENERATIVE AI

In this work, we used generative AI tools to assist with language polishing and presentation, including improving grammar, clarity, readability, academic expression, and the organization of selected passages. The scientific ideas, methodological design, experiments, data analysis, and conclusions were developed and validated by the authors. All AI-assisted content was carefully reviewed, verified, and revised by the authors, who take full responsibility for the accuracy, originality, and final content of the manuscript.

ETHICS STATEMENT

This study is a computational investigation based exclusively on previously released public TCR–epitope datasets. We did not collect new human or animal data, interact with research participants, or access direct personal identifiers. Ethical approval, informed consent, and data-governance requirements associated with the original data collection remain governed by the corresponding source studies. We used the datasets only for the research purposes and evaluation protocols described in this work and did not attempt to re-identify any individual. The proposed method is intended for research use rather than direct clinical decision-making. Its predictions may be affected by biases in source datasets, including uneven epitope frequencies, repertoire composition, negative-sample construction, and population coverage. Consequently, biological or therapeutic conclusions should not be drawn from model predictions without appropriate experimental validation. The authors take responsibility for the integrity, interpretation, and reporting of this work.

REPRODUCIBILITY STATEMENT

We provide the information required to reproduce the experiments throughout the main text and appendices. The paper specifies the problem formulation, model architecture, training objectives, inference procedure, datasets, preprocessing and input-construction steps, evaluation metrics, baseline configurations, and experimental protocols. Detailed architectural and optimization settings, including sequence lengths, representation dimensions, convolutional configurations, loss coefficients, learning rates, batch size, training duration, random seed, and stochastic weight averaging, are reported in Appendix B. Dataset descriptions, partitioning strategies, and evaluation procedures are provided in Appendix C, while baseline configurations and additional experimental results are documented in the corresponding appendices. All reported methods are evaluated under consistent dataset splits and metric definitions to facilitate direct comparison.

REFERENCES

Giancarlo Croce, Sara Bobisse, Dana Léa Moreno, Julien Schmidt, Philippe Guillame, Alexandre Harari, and David Gfeller. Deep learning predictions of tcr-epitope interactions reveal epitope-specific chains in dual alpha t cells. *Nature Communications*, 15(1):3211, 2024.

Pradyot Dash, Andrew J. Fiore-Gartland, Tomer Hertz, George C. Wang, Shalini Sharma, Aisha Souquette, Jeremy Chase Crawford, E. Bridie Clemens, Thi H. O. Nguyen, Katherine Kedzierska, Nicole L. La Gruta, Philip Bradley, and Paul G. Thomas. Quantifiable predictive features define epitope-specific t cell receptor repertoires. *Nature*, 547:89–93, 2017.

Lihua Deng, Cedric Ly, Sina Abdollahi, Yu Zhao, Immo Prinz, and Stefan Bonn. Performance comparison of tcr-pmhc prediction tools reveals a strong data dependency. *Frontiers in Immunology*, 14:1128326, 2023.

Felix Drost, Anna Chernysheva, Mahmoud Albahah, Katharina Kocher, Kilian Schober, and Benjamin Schubert. Benchmarking of t cell receptor-epitope predictors with epytope-tcr. *Cell Genomics*, 5(8):100946, 2025.

Xikang Feng, Miaozhe Huo, He Li, Yongze Yang, Yuepeng Jiang, Liang He, and Shuai Cheng Li. A comprehensive benchmarking for evaluating tcr embeddings in modeling tcr-epitope interactions. *Briefings in Bioinformatics*, 26(1):bbaf030, 2025.

Xiangzheng Fu, Li Peng, Haowen Chen, Mingqiang Rong, Yifan Chen, Dongsheng Cao, Sisi Yuan, and Aiping Lu. GRAPE: graph-regularized protein language modeling unlocks tcr-epitope binding specificity. *Briefings in Bioinformatics*, 26(5):bbaf522, 2025.

Yicheng Gao, Yuli Gao, Yuxiao Fan, Chengyu Zhu, Zhiting Wei, Chi Zhou, Guohui Chuai, Qinchang Chen, He Zhang, and Qi Liu. Pan-peptide meta learning for t-cell receptor-antigen binding recognition. *Nature Machine Intelligence*, 5:236–249, 2023.

Zahra S. Ghoreyshi, Noah Tubo, Luca Zammataro, Xizeng Mao, Ho Ngai, Duncheng Wang, Yibin Chen, Qiuming He, Eduardo Cisneros de la Rosa, Shoudan Liang, Priya J. Koppikar, Xingcheng Lin, Jeffrey J. Molldrem, and Jason T. George. Biophysical modeling for accurate T cell specificity prediction of viral and tumor antigens. *Nature Communications*, 17(1):7885, 2026.

Jacob Glanville, Huang Huang, Allison Nau, Olivia Hatton, Lisa E. Wagar, Florian Rubelt, Xuhuai Ji, Arnold Han, Sheri M. Krams, Christina Pettus, Nikhil Haas, Cecilia S. Lindestam Arlehamn, Alessandro Sette, Scott D. Boyd, Thomas J. Scriba, Olivia M. Martinez, and Mark M. Davis. Identifying specificity groups in the t cell receptor repertoire. *Nature*, 547:94–98, 2017.

Filippo Grazioli, Pierre Machart, Anja Mösch, Kai Li, Leonardo V Castorina, Nico Pfeifer, and Martin Renqiang Min. Attentive Variational Information Bottleneck for TCR–peptide interaction prediction. *Bioinformatics*, 39(1):btac820, 2023.

Mengnan Jiang, Zilan Yu, and Xun Lan. VitTCR: A deep learning method for peptide recognition prediction. *iScience*, 27(5):109770, 2024.

Yuepeng Jiang, Miaozhe Huo, and Shuai Cheng Li. TEINet: a deep learning framework for prediction of tcr-epitope binding specificity. *Briefings in Bioinformatics*, 24(2):bbad086, 2023.

Emmi Jokinen, Alexandru Dumitrescu, Jani Huuhtanen, Vladimir Gligorijevic, Satu Mustjoki, Richard Bonneau, Markus Heinonen, and Harri Lahdesmaki. TCRconv: predicting recognition between t cell receptors and epitopes using contextualized motifs. *Bioinformatics*, 39(1):btac788, 2023.

Ha Young Kim, Sungsik Kim, Woong-Yang Park, and Dongsup Kim. TSpred: a robust prediction framework for tcr–epitope interactions using paired chain tcr sequence data. *Bioinformatics*, 40 (8):btae472, 2024.

Dani Korpela, Emmi Jokinen, Alexandru Dumitrescu, Jani Huuhtanen, Satu Mustjoki, and Harri Lähdesmäki. EPIC-TRACE: predicting tcr binding to unseen epitopes using attention and contextualized embeddings. *Bioinformatics*, 39(12):btad743, 2023.

Jiarui Li, Zixiang Yin, Zhengming Ding, Samuel J Landry, and Ramgopal R. Mettu. Rational Multi-Modal Transformers for TCR-pMHC Prediction. In *Proceedings of the 16th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pp. 1–8. Association for Computing Machinery, 2025.

Tianshi Lu, Ze Zhang, James Zhu, Yunguan Wang, Peixin Jiang, Xue Xiao, Chantale Bernatchez, John V. Heymach, Don L. Gibbons, Jun Wang, Lin Xu, Alexandre Reuben, and Tao Wang. Deep learning-based prediction of the t cell receptor-antigen binding specificity. *Nature Machine Intelligence*, 3:864–875, 2021.

Yanping Lu, Yuyan Wang, Meng Xu, Bingbing Xie, Yumeng Yang, Haodong Xu, and Shengbao Suo. Assessment of computational methods in predicting TCR–epitope binding recognition. *Nature Methods*, 23(1):248–259, 2026.

Pieter Meysman, Justin Barton, Barbara Bravi, Liel Cohen-Lavi, Vadim Karnaukhov, Elias Lilleskov, Alessandro Montemurro, Morten Nielsen, Thierry Mora, Paul Pereira, Anna Postovskaya, María Rodríguez Martínez, Jorge Fernandez-de Cossio-Diaz, Alexandra Vujkovic, Aleksandra M. Walczak, Anna Weber, Rose Yin, Anne Eugster, and Virag Sharma. Benchmarking solutions to the T-cell receptor epitope prediction problem: IMMREP22 workshop report. *ImmunoInformatics*, 9:100024, 2023.

Alessandro Montemurro, Viktoria Schuster, Helle Rus Povlsen, Amalie Kai Bentzen, Vanessa Jurtz, William D. Chronister, Austin Crinklaw, Sine R. Hadrup, Ole Winther, Bjoern Peters, Leon Eyrich Jessen, and Morten Nielsen. NetTCR-2.0 enables accurate prediction of tcr-peptide binding by using paired tcr alpha and beta sequence data. *Communications Biology*, 4:1060, 2021.

Morten Nielsen, Anne Eugster, Mathias Fynbo Jensen, Manisha Goel, Andreas Tiffeau-Mayer, Aurelien Pelissier, Sebastiaan Valkiers, María Rodríguez Martínez, Barthélémy Meynard-Piganeeau, Victor Greiff, Thierry Mora, Aleksandra M. Walczak, Giancarlo Croce, Dana L. Moreno, David Gfeller, Pieter Meysman, and Justin Barton. Lessons learned from the IMMREP23 TCR-epitope prediction challenge. *ImmunoInformatics*, 16:100045, 2024.

My-Diem Nguyen Pham, Thanh-Nhan Nguyen, Le Son Tran, Que-Tran Bui Nguyen, Thien-Phuc Hoang Nguyen, Thi Mong Quynh Pham, Hoai-Nghia Nguyen, Hoa Giang, Minh-Duy Phan, and Vy Nguyen. epiTCR: a highly sensitive predictor for tcr-peptide binding. *Bioinformatics*, 39 (5):btad284, 2023.

Furong Qi, Qiang Huang, Yao Xuan, Yingyin Cao, Yunyun Shen, Yihan Ren, Zhe Liu, and Zheng Zhang. A roadmap for t cell receptor-peptide-bound major histocompatibility complex binding prediction by machine learning: glimpse and foresight. *Briefings in Bioinformatics*, 26(4): bbaf327, 2025.

Se Yeon Seo and Je-Keun Rhee. TCR-epiDiff: solving dual challenges of tcr generation and binding prediction. *Bioinformatics*, 41(Supplement_1):i125–i132, 2025.

Ido Springer, Hadar Besser, Nili Tickotsky-Moskovitz, Shirit Dvorkin, and Yoram Louzoun. ERGO: a deep learning model for predicting t cell receptor antigen specificity. *Bioinformatics*, 36 (Supplement_1):i262–i270, 2020.

Rajitha Rajeshwar T., Omar N. A. Demerdash, and Jeremy C. Smith. TCR-H: explainable machine learning prediction of T-cell receptor epitope binding on unseen datasets. *Frontiers in Immunology*, 15:1426173, 2024.

Varun Ullanat, Bowen Jing, Samuel Sledzieski, and Bonnie Berger. Learning the language of protein-protein interactions. *Nature Communications*, 17:1199, 2026.

Meng Wang, Wei Fan, Tianrui Wu, and Min Li. TPepRet: a deep learning model for characterizing T-cell receptors–antigen binding patterns. *Bioinformatics*, 41(1):btaf022, 2025.

Anna Weber, Jannis Born, and Maria Rodriguez Martinez. TITAN: T-cell receptor specificity prediction with bimodal attention networks. *Bioinformatics*, 37(Supplement_1):i237–i244, 2021.

Anna Weber, Aurélien Pélissier, and María Rodríguez Martínez. T-cell receptor binding prediction: A machine learning revolution. *ImmunoInformatics*, 15:100040, 2024.

Jie Wu, Meng Qi, Feiyan Zhang, and Yuanjie Zheng. TPBTE: A model based on convolutional Transformer for predicting the binding of TCR to epitope. *Molecular Immunology*, 157:30–41, 2023.

Kevin E. Wu, Kathryn Yost, Bence Daniel, Julia Belk, Yu Xia, Takeshi Egawa, Ansuman Satpathy, Howard Chang, and James Zou. TCR-BERT: learning the grammar of t-cell receptors for flexible antigen-binding analyses. In *Proceedings of Machine Learning Research*, volume 240, pp. 194–229, 2024.

Chenpeng Yu, Xing Fang, Shiye Tian, Judong Luo, and Hui Liu. A lightweight TcrLM model predicts T cell receptor and epitope binding specificity. *Cell Reports Methods*, 6(5):101378, 2026.

Zilan Yu, Mengnan Jiang, and Xun Lan. HeteroTCR: A heterogeneous graph neural network-based method for predicting peptide-tcr interaction. *Communications Biology*, 7:684, 2024.

Yajing Yuan, Junwei Chen, Yufang Zhang, Yitian Fang, Zhongcheng Fang, Yanyi Chu, Jiayi Li, Chen Zhang, Yuzhe Li, and Dongqing Wei. A Biologically Informed Vision-Guided Framework for Interpretable T Cell Receptor–Epitope Binding Prediction. *Advanced Science*, 13(4):e12544, 2026.

A Related Work

Existing computational methods for TCR–epitope specificity prediction can be organized into four directions according to where representation capacity is allocated and how TCR–epitope compatibility is learned: TCR-centric sequence modeling, pairwise representation and interaction learning, pretrained and topology-enhanced modeling, and generalization-oriented learning and evaluation.

A.1 TCR-Centric Sequence and Paired-Chain Modeling

TCR-centric methods establish that antigen specificity can be partially inferred from recurrent sequence patterns in the receptor repertoire. Early studies such as TCRdist and GLIPH characterized epitope-associated TCRs through sequence similarity and enriched CDR3 motifs Dash et al. (2017); Glanville et al. (2017). Subsequent deep models replaced fixed similarity functions with learnable sequence representations. NetTCR-2.0 demonstrated the value of jointly modeling paired α- and β-chains Montemurro et al. (2021), while TCRconv combined contextualized residue features with convolutional motif extraction Jokinen et al. (2023). More recently, MixTCRpred constructed epitope-specific predictors from a curated paired-chain repertoire and showed that both chains can contribute to antigen recognition Croce et al. (2024). Collectively, these studies demonstrate that local motifs, surrounding sequence context, and paired-chain organization provide complementary information for representing TCR specificity.

These approaches primarily strengthen receptor-side representation. Similarity-based methods remain dependent on previously observed sequence neighborhoods, whereas epitope-specific methods such as MixTCRpred train separate predictors for individual targets. They therefore do not generally formulate specificity prediction as a unified mapping from a structured paired-TCR representation to a shared finite candidate space.

A.2 Pairwise Representation and Interaction Learning

Pairwise representation methods explicitly encode both TCR and epitope sequences and learn their compatibility in a joint prediction model. ERGO introduced recurrent encoders for TCR and peptide sequences Springer et al. (2020), while TITAN employed bimodal attention to exchange information between the two representations Weber et al. (2021). Recent extensions have explored transfer learning in TEINet Jiang et al. (2023), specialized neural sequence encoding in epiTCR Pham et al. (2023), variational information bottlenecks in vibtcr Grazioli et al. (2023), and convolutional or vision Transformer architectures in TPBTE and VitTCR Wu et al. (2023); Jiang et al. (2024). TPepRet combines recurrent sequence modeling with retention mechanisms Wang et al. (2025), whereas TSpred integrates paired-chain convolutional encoding with reciprocal attention for pan-specific prediction Kim et al. (2024). TCR-Rational further employs a multimodal Transformer design to model TCR–pMHC interactions Li et al. (2025).

Explicit pairwise modeling is well suited to settings in which the epitope sequence must remain a model input, particularly pan-specific or unseen-epitope prediction. However, it generally requires separately learning TCR and epitope representations and subsequently aligning them through concatenation, attention, or cross-sequence interaction. This introduces an additional representation-alignment problem whose necessity depends on the intended prediction setting.

A.3 Pretrained, Graph, and Generative Modeling

Recent methods enrich TCR–epitope prediction with pretrained sequence priors, relational topology, and generative learning. pMTnet transfers representations learned from large protein sequence collections to TCR–antigen specificity prediction Lu et al. (2021), while TCR-BERT learns contextualized TCR representations through self-supervised pretraining Wu et al. (2024). HeteroTCR represents known TCR–epitope relations as a heterogeneous graph Yu et al. (2024), and GRAPE combines protein language model embeddings with graph regularization and imbalance-aware optimization Fu et al. (2025).

The modeling scope has also expanded beyond conventional discriminative prediction. TCR-epiDiff combines epitope-conditioned diffusion with TCR generation and binding prediction Seo & Rhee (2025), tcrLM investigates lightweight language modeling for TCR–epitope specificity Yu et al.

(2026), and DAISY introduces biologically informed visual representations for interpretable interaction prediction Yuan et al. (2026). These developments provide richer sequence priors, interaction structures, and interpretability mechanisms. Nevertheless, most retain explicit representations for both molecular sides or model their interactions through increasingly elaborate latent structures.

A.4 Generalization-Oriented Learning and Benchmarking

Generalization beyond frequently observed epitopes has become a central focus of recent research. PanPep employs meta-learning to transfer specificity knowledge across peptides Gao et al. (2023), EPIC-TRACE combines contextualized embeddings with attention for unseen-epitope prediction Korpela et al. (2023), and TCR-H uses interpretable physicochemical representations under increasingly stringent unseen-data settings T. et al. (2024). These methods seek to improve transfer beyond the TCR–epitope combinations that dominate the training distribution.

In parallel, systematic benchmarks have shown that measured performance depends strongly on both data construction and evaluation protocol. IMMREP22 and IMMREP23 evaluated prediction under standardized paired-chain settings Meysman et al. (2023); Nielsen et al. (2024), while cross-method analyses identified substantial dependencies on training data and peptide frequency Deng et al. (2023). More recent studies have evaluated pretrained predictors through ePytope-TCR Drost et al. (2025), compared alternative TCR embedding strategies Feng et al. (2025), and assessed a large collection of model configurations across heterogeneous datasets Lu et al. (2026). These evaluations consistently indicate that negative-sample construction, sequence overlap, epitope abundance, training-set composition, and independent test design can substantially affect the reported results. They further show that increasing encoder or interaction complexity alone does not ensure robust cross-dataset or unseen-epitope generalization.

Taken together, existing studies address two related but distinct prediction regimes. Pan-specific and open-set prediction typically require the epitope sequence to remain an explicit model input so that the predictor can extrapolate to previously unseen targets. In finite-candidate classification, however, the candidate epitopes form a fixed output vocabulary and primarily define the identities to be discriminated. The representational asymmetry considered in this work is therefore task-conditioned rather than a claim that epitopes are biologically simple.

Based on this distinction, AsymTCR allocates structured representation capacity to the diverse paired TCR chains while modeling candidate epitopes as discrete categories in a shared prediction space. Unlike dual-encoder approaches that align two continuous representation spaces, or epitope-specific approaches that train a separate predictor for each target, AsymTCR learns a unified discriminative mapping from contextualized paired-TCR representations to all candidate epitopes.

B Implementation and Hyperparameter Settings

AsymTCR was implemented with a frozen ProtBERT backbone for contextual sequence encoding. The two TCR chains use independently parameterized global projections, while the multiscale local convolutional branch shares its parameters across the α- and β-chains. All task-specific components were trained jointly using the objective defined in Section 2.2.3. Table 2 summarizes the architectural, optimization, and objective-specific settings used in all experiments.

Table 2: Implementation and hyperparameter settings of AsymTCR.

Category	Setting	Value
Architecture	Protein encoder	Frozen ProtBERT
	ProtBERT hidden dimension	1024
	Padded CDR3 length	32
	Convolution kernel sizes	$3, 5, 9, 15, 21$
	Channels per scale	$120, 120, 100, 80, 60$
	Global representation dimension	384 per chain
	Local representation dimension	100 per chain
	Paired-TCR representation dimension	968
Optimization	Optimizer	Adam
	Adam coefficients	$\beta_1 = 0.9,\ \beta_2 = 0.999$
	Learning rate	2×10^{-4} (conv.), 3×10^{-3} (linear)
	Weight decay	10^{-5}
	Gradient clipping	Maximum norm 5
	Batch size / epochs	$512/270$
	Learning-rate schedule	Cosine annealing
Regularization	Dropout	0.1
	Random seed	60
	Stochastic weight averaging	From epoch 194
	Batch statistics	Refreshed after SWA
Objective	Negative focusing exponent	$\gamma = 2$
	Ranking margin	$m = 0.5$
	Pair-loss coefficient	$\lambda_{\text{pair}} = 0.15$
	Ranking-loss coefficient	$\lambda_{\text{rank}} = 0.10$

C Dataset Details

NM-Benchmark. The CDR3β+others benchmark released by Lu et al. (2026) was used as our primary benchmark. After preprocessing, its training cohort contained 4,224 TCRs associated with 57 epitopes. We constructed 4,154 positive and 4,154 negative TCR–epitope pairs and followed a five-fold cross-validation protocol and retained held-out predictions from each fold. The comparison table reports the mean and sample standard deviation of the five fold-level metrics; the ablation and parameter-sensitivity figures compute metrics once from pooled out-of-fold (OOF) predictions. The accompanying independent seen-epitope test set contained 330 retained TCRs covering 18 epitopes, resulting in 313 positive and 313 negative pairs. The comparison table evaluates the five trained models separately and reports the mean and sample standard deviation of their metrics. The independent-set ablation and parameter-sensitivity figures instead use the mean probability of the five models.

GSE332840. GSE332840 was released by Ghoreyshi et al. (2026) and contains antigen-specific single-cell TCR sequencing data. After applying the sequence requirements of our model and retaining epitopes with sufficient usable receptors, 47 TCRs associated with two epitopes were retained. The resulting balanced subset contained 47 positive and 47 negative TCR–epitope pairs and was evaluated using five-fold cross-validation with held-out predictions retained from every fold.

IMMREP22. IMMREP22 was introduced by Meysman et al. (2023) in the first public IMMREP TCR–epitope prediction benchmark. After preprocessing, 8,229 TCRs covering 17 epitopes were

retained. The resulting dataset contained 3,064 positive and 15,312 negative TCR–epitope pairs. Five-fold cross-validation was performed, and held-out predictions were retained from every fold.

IMMREP23. IMMREP23 is a subsequent TCR specificity benchmark introduced by Nielsen et al. (2024). After preprocessing, 727 TCRs covering 20 epitopes were retained, yielding 598 positive and 2,886 negative TCR–epitope pairs. Model performance was evaluated using five-fold cross-validation with held-out predictions retained from every fold. Table 3 summarizes the datasets used in our experiments after preprocessing. "Used TCRs" denotes the number of TCRs retained after sequence filtering, whereas the positive and negative columns report the numbers of TCR–epitope pairs used for training or evaluation.

Table 3: Detailed statistics of the datasets used in the experiments.

Dataset	No. of epitopes	Used TCRs	Positive / Negative	Evaluation
NM-Benchmark: training cohort	57	4,224	4,154 / 4,154	Five-fold CV
NM-Benchmark: independent test	18	330	313 / 313	Five models evaluated separately
GSE332840-derived subset	2	47	47 / 47	Five-fold CV
IMMREP22	17	8,229	3,064 / 15,312	Five-fold CV
IMMREP23	20	727	598 / 2,886	Five-fold CV

D Evaluation Metrics

We evaluated prediction performance using AUPRC, AUROC, and ACC. Let TP, TN, FP, and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively. Precision and recall are defined as

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \qquad \mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}. \tag{21}$$

The area under the precision–recall curve is computed as

$$\mathrm{AUPRC} = \int_0^1 \mathrm{Precision}(\mathrm{Recall})\, d\,\mathrm{Recall}, \tag{22}$$

which summarizes the trade-off between precision and recall over different decision thresholds.

For AUROC, the true-positive rate (TPR) and false-positive rate (FPR) are defined as

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \qquad \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}}. \tag{23}$$

The area under the receiver operating characteristic curve is then

$$\mathrm{AUROC} = \int_0^1 \mathrm{TPR}(\mathrm{FPR})\, d\,\mathrm{FPR}. \tag{24}$$

Accuracy measures the proportion of correctly classified samples:

$$\mathrm{ACC} = \frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}. \tag{25}$$

AUPRC and AUROC are computed by varying the decision threshold over the predicted specificity scores, whereas ACC is calculated from the final binary predictions under the adopted decision rule.

E Details of Comparison Methods

We considered 16 baseline configurations corresponding to 13 published methods. When a method provided multiple input settings, the configurations relevant to our experiments were retained as separate baselines. The main characteristics of these methods are described below.

TCRconv. TCRconv Jokinen et al. (2023) combines contextual residue representations extracted by a pretrained protein language model with convolutional operations designed to capture local recognition-related sequence motifs. We evaluated three configurations: TCRconv-fullAB uses paired α- and β-chain information, whereas TCRconv-fullA and TCRconv-fullB use the individual α- and β-chains, respectively. These configurations provide both paired-chain and single-chain contextual baselines.

TCR-Rational. TCR-Rational Li et al. (2025) employs a rationally designed multimodal Transformer architecture for TCR–pMHC prediction, with its model design informed by interpretability analyses of different biological input modalities. Following the available implementation, we used the `cdr3_only_binder` configuration in our comparison.

epiTCR. epiTCR Pham et al. (2023) predicts TCR–peptide binding by combining sequence-derived representations with a random-forest classifier. We used the original epiTCR formulation rather than its later epiTCR-KDA variant.

VitTCR. VitTCR Jiang et al. (2024) transforms paired TCR–peptide information into Atchley-factor-based two-dimensional representations (AtchleyMaps) and applies a Vision Transformer to learn interaction patterns for peptide recognition.

vibtcr. vibtcr Grazioli et al. (2023) introduces an attentive variational information bottleneck framework for TCR–peptide interaction prediction. We retained both the paired-chain vibtcr-AB configuration and the corresponding single-chain vibtcr configuration as separate baselines.

TPepRet. TPepRet Wang et al. (2025) integrates amino-acid physicochemical properties, bidirectional gated recurrent units, and RetNet to model the relationship between CDR3β sequences and antigen peptides.

MINT. MINT Ullanat et al. (2026) is based on the Multimeric INteraction Transformer, a protein interaction model pretrained to learn general sequence-level protein–protein interaction patterns. Its learned interaction representations can be adapted to TCR–epitope specificity prediction.

TEINet. TEINet Jiang et al. (2023) is a deep learning framework for TCR–epitope binding specificity prediction that leverages transfer-based sequence representation learning to model the relationship between TCR and epitope sequences.

DAISY. DAISY Yuan et al. (2026) is a biologically informed framework that combines local interaction representations with global physicochemical features and employs a conditional adaptive fusion mechanism for interpretable TCR–epitope binding prediction.

TPBTE. TPBTE Wu et al. (2023) employs a convolutional Transformer to capture both local and global sequence information and further models TCR–epitope interactions through cross-attention.

tcrLM. tcrLM Yu et al. (2026) is a lightweight TCR language model pretrained on large-scale CDR3 sequences and subsequently adapted to TCR–epitope binding-specificity prediction.

TCR-BERT. TCR-BERT Wu et al. (2024) learns self-supervised language representations of TCR sequences and transfers the resulting representations to downstream antigen-binding analyses.

TCR-H. TCR-H T. et al. (2024) represents TCR and epitope sequences using physicochemical descriptors and performs explainable binding prediction using conventional machine-learning models, with the final reported model based on a support vector machine.

F Complete Comparison Results

Table 4: Performance on GSE332840. Values are mean ± standard deviation.

Method	AUPRC	AUROC	ACC
AsymTCR	0.942 ± 0.055	0.935 ± 0.061	0.853 ± 0.109
TCRconv-fullAB	$\mathbf{0.954 \pm 0.048}$	$\underline{0.944 \pm 0.058}$	$\mathbf{0.876 \pm 0.114}$
TCRconv-fullA	0.920 ± 0.103	0.906 ± 0.122	0.798 ± 0.186
TCRconv-fullB	0.867 ± 0.121	0.887 ± 0.089	0.831 ± 0.054
epiTCR	0.748 ± 0.075	0.709 ± 0.075	0.566 ± 0.048
vibtcr-AB	0.727 ± 0.143	0.663 ± 0.177	0.581 ± 0.150
VitTCR	0.729 ± 0.197	0.701 ± 0.196	0.668 ± 0.177
TCR-BERT	0.496 ± 0.193	0.467 ± 0.220	0.446 ± 0.133
TPepRet	0.688 ± 0.184	0.672 ± 0.193	0.591 ± 0.284
vibtcr	0.638 ± 0.092	0.629 ± 0.080	0.512 ± 0.076
TCR-Rational	$\underline{0.953 \pm 0.055}$	$\mathbf{0.946 \pm 0.062}$	$\underline{0.863 \pm 0.095}$
TEINet	0.610 ± 0.127	0.543 ± 0.115	0.500 ± 0.000
TCR-H	0.373 ± 0.063	0.254 ± 0.158	0.364 ± 0.148
DAISY	0.538 ± 0.099	0.526 ± 0.089	0.500 ± 0.000
MINT	0.650 ± 0.058	0.543 ± 0.095	0.500 ± 0.000
TPBTE	0.567 ± 0.090	0.528 ± 0.038	0.500 ± 0.000
tcrLM	0.526 ± 0.025	0.542 ± 0.088	0.525 ± 0.071

Mean and sample SD across five held-out folds. Native method coverage is retained. Bold: best; underline: second best.

Table 5: Performance on IMMREP22. Values are mean ± standard deviation.

Method	AUPRC	AUROC	ACC
AsymTCR	$\underline{0.694 \pm 0.082}$	$\mathbf{0.842 \pm 0.039}$	$\underline{0.898 \pm 0.015}$
TCRconv-fullAB	$\mathbf{0.697 \pm 0.077}$	$\underline{0.830 \pm 0.043}$	$\mathbf{0.901 \pm 0.012}$
TCRconv-fullA	0.600 ± 0.073	0.778 ± 0.039	0.882 ± 0.006
TCRconv-fullB	0.623 ± 0.098	0.780 ± 0.053	0.892 ± 0.017
epiTCR	0.580 ± 0.095	0.777 ± 0.039	0.745 ± 0.008
vibtcr-AB	0.582 ± 0.101	0.783 ± 0.052	0.836 ± 0.033
VitTCR	0.438 ± 0.132	0.695 ± 0.063	0.846 ± 0.014
TCR-BERT	0.556 ± 0.099	0.764 ± 0.045	0.875 ± 0.009
TPepRet	0.481 ± 0.074	0.716 ± 0.042	0.845 ± 0.008
vibtcr	0.494 ± 0.116	0.743 ± 0.054	0.764 ± 0.023
TCR-Rational	0.164 ± 0.005	0.472 ± 0.040	0.834 ± 0.014
TEINet	0.405 ± 0.114	0.705 ± 0.056	0.715 ± 0.031
TCR-H	0.364 ± 0.031	0.698 ± 0.008	0.838 ± 0.018
DAISY	0.299 ± 0.056	0.646 ± 0.036	0.531 ± 0.065
MINT	0.155 ± 0.012	0.438 ± 0.070	0.829 ± 0.014
TPBTE	0.182 ± 0.008	0.537 ± 0.012	0.834 ± 0.014
tcrLM	0.162 ± 0.019	0.469 ± 0.095	0.740 ± 0.168

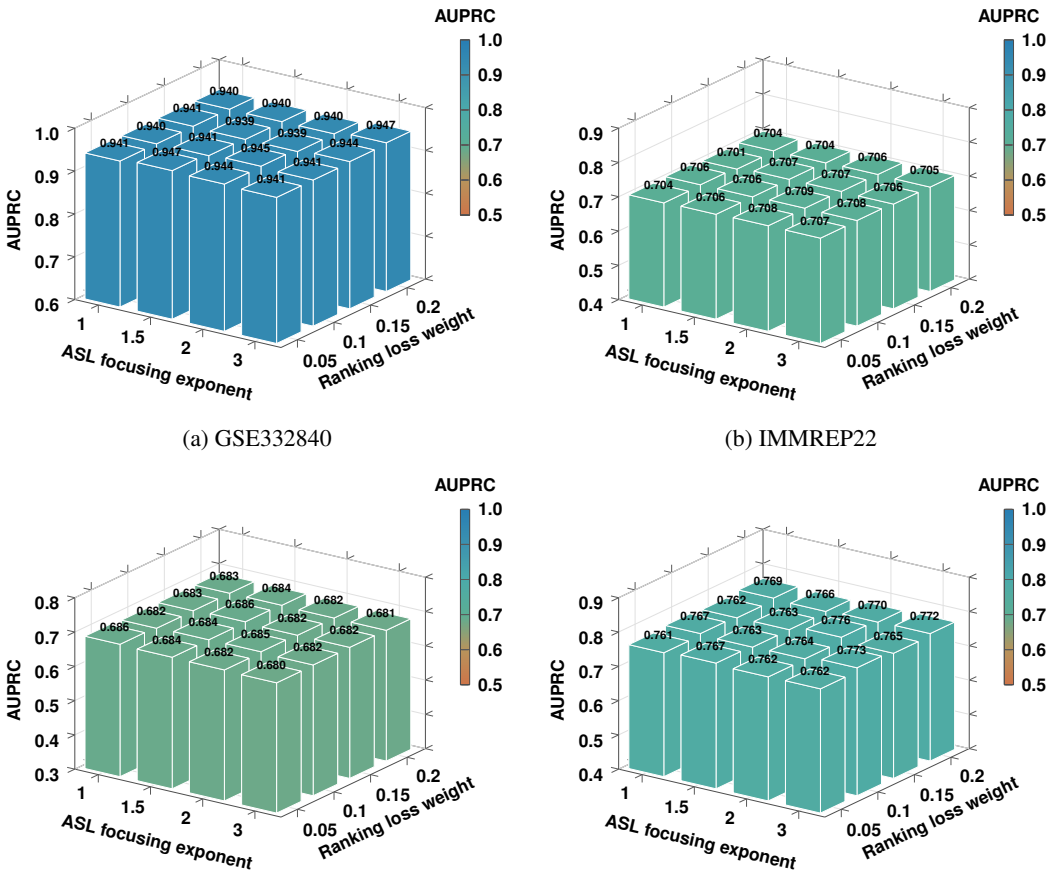
Mean and sample SD across five held-out folds. Native method coverage is retained. Bold: best; underline: second best.

Table 6: Performance on IMMREP23. Values are mean ± standard deviation.

Method	AUPRC	AUROC	ACC
AsymTCR	$\underline{0.676 \pm 0.090}$	$\mathbf{0.818 \pm 0.051}$	$\mathbf{0.898 \pm 0.006}$
TCRconv-fullAB	$\mathbf{0.683 \pm 0.079}$	$\underline{0.802 \pm 0.042}$	$\underline{0.895 \pm 0.006}$
TCRconv-fullA	0.606 ± 0.066	0.760 ± 0.039	0.883 ± 0.008
TCRconv-fullB	0.621 ± 0.071	0.779 ± 0.022	0.884 ± 0.016
epiTCR	0.512 ± 0.085	0.727 ± 0.037	0.607 ± 0.035
vibtcr-AB	0.390 ± 0.112	0.689 ± 0.051	0.613 ± 0.125
VitTCR	0.415 ± 0.087	0.688 ± 0.027	0.811 ± 0.028
TCR-BERT	0.424 ± 0.105	0.689 ± 0.044	0.850 ± 0.020
TPepRet	0.390 ± 0.101	0.702 ± 0.020	0.785 ± 0.019
vibtcr	0.351 ± 0.094	0.636 ± 0.021	0.515 ± 0.136
TCR-Rational	0.176 ± 0.031	0.493 ± 0.046	0.827 ± 0.025
TEINet	0.251 ± 0.067	0.566 ± 0.066	0.372 ± 0.133
TCR-H	0.317 ± 0.042	0.651 ± 0.032	0.827 ± 0.027
DAISY	0.205 ± 0.017	0.549 ± 0.037	0.634 ± 0.248
MINT	0.178 ± 0.019	0.440 ± 0.037	0.318 ± 0.275
TPBTE	0.167 ± 0.017	0.484 ± 0.035	0.827 ± 0.025
tcrLM	0.184 ± 0.023	0.519 ± 0.026	0.757 ± 0.114

Mean and sample SD across five held-out folds. Native method coverage is retained. Bold: best; underline: second best.

G Additional Parameter Sensitivity Results



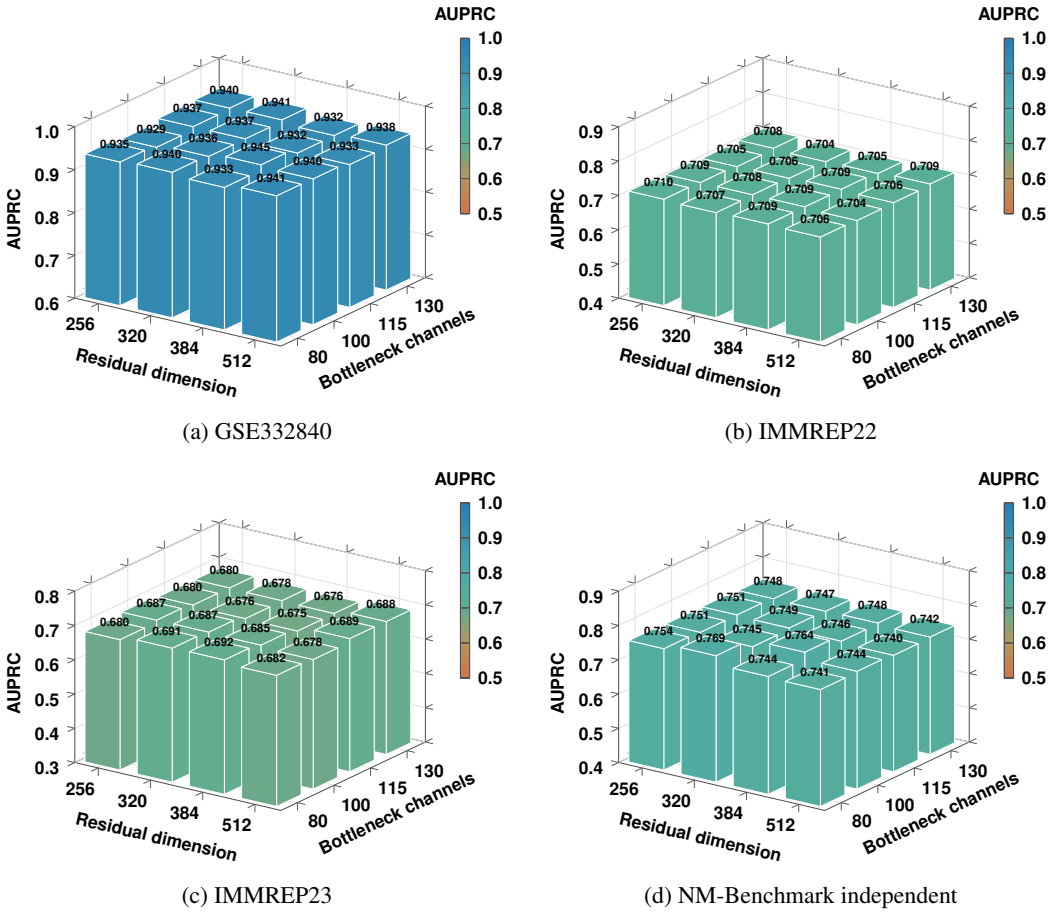
(a) GSE332840

(b) IMMREP22

(c) IMMREP23

(d) NM-Benchmark independent

Figure 6: **Additional loss-parameter sensitivity results.** Each panel varies the asymmetric-loss focusing exponent and ranking-loss weight over 16 measured combinations. The first three use pooled five-fold out-of-fold AUPRC; the independent panel uses the mean probability of five models. Bars show measured values. Truncated vertical axes and the shared colour scale follow Figure 4. Independent-test values are descriptive rather than a basis for selecting hyperparameters.



(a) GSE332840

(b) IMMREP22

(c) IMMREP23

(d) NM-Benchmark independent

Figure 7: **Additional architecture-parameter sensitivity results.** Each panel varies the residual dimension and bottleneck channels over 16 measured combinations. The first three use pooled five-fold out-of-fold AUPRC; the independent panel uses the mean probability of five models. Bars show measured values. Truncated vertical axes and the shared colour scale follow Figure 4. Independent-test values are descriptive rather than a basis for selecting hyperparameters.

H Ablation Studies

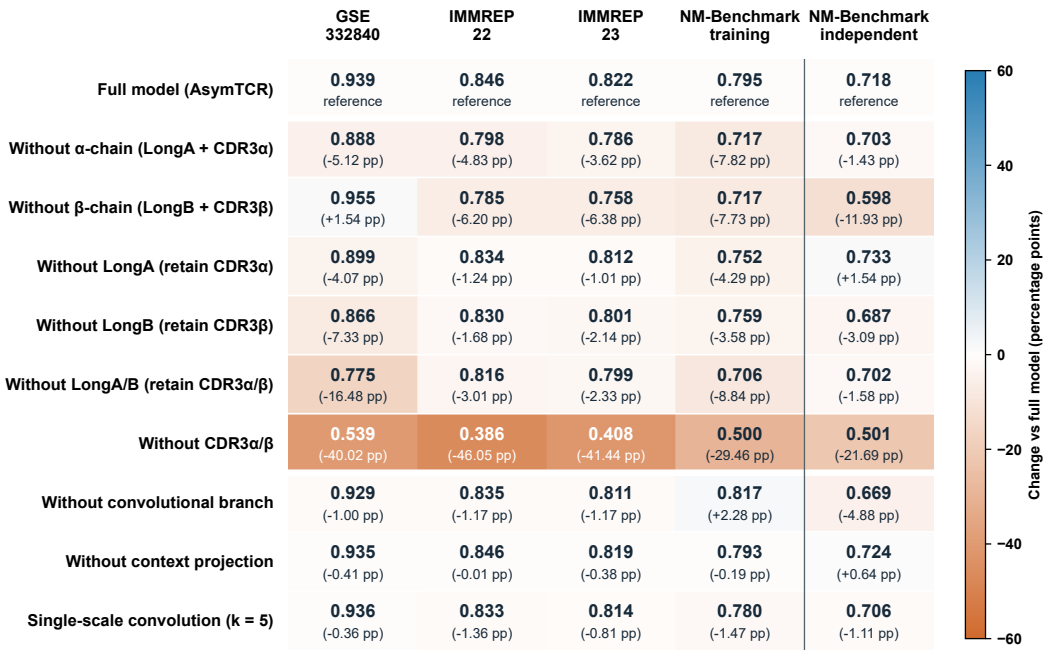


Figure 8: Input and representation ablation results measured by AUROC. The full model is compared with nine distinct input and representation ablations. Each cell shows the absolute score and, in parentheses, the change from the full model in percentage points (ablation minus full model). Orange indicates a decrease and blue an increase. The first four settings, including NM-Benchmark training, use pooled five-fold OOF predictions; NM-Benchmark independent (test626) uses the mean probability of five models. These estimates are distinct from fold-level metric means.

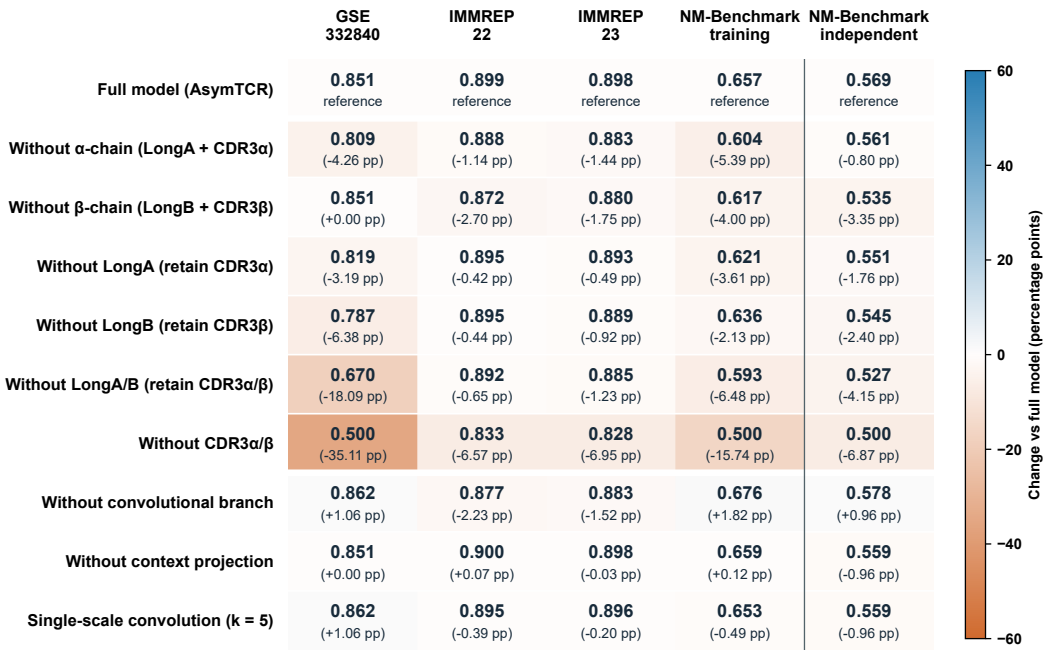


Figure 9: Input and representation ablation results measured by ACC. The full model is compared with nine distinct input and representation ablations. Each cell shows the absolute score and, in parentheses, the change from the full model in percentage points (ablation minus full model). Orange indicates a decrease and blue an increase. The first four settings, including NM-Benchmark training, use pooled five-fold OOF predictions; NM-Benchmark independent (test626) uses the mean probability of five models. These estimates are distinct from fold-level metric means.