EXPLAINABILITY FROM TRAINING WITH APPLICATIONS TO TCR-EPITOPE PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep learning models have achieved strong performance in artificial intelligence for science, yet their black-box nature limits our understanding of how they learn scientific tasks. Existing methods for interpretability provide limited insight into how models organize evidence and evolve during learning. We introduce Explainability From Training (EFT), a model-agnostic paradigm that traces model interpretation during training to explain why models rely on specific features and how they organize these features as predictive evidence. We apply EFT to four state-of-the-art T cell receptor (TCR)-epitope prediction models, TCR-SRIM, TULIP, MixTCRpred, and NetTCR-2.2, spanning post-hoc and interpret-by-design approaches as well as transformers and CNNs. To investigate how structural information affects model explanations, we introduce a benchmark, TCR-XAI2, containing 388 unique experimentally resolved TCR-epitope structures, complemented by structures predicted using AlphaFold3, Boltz-2, TCRModel2, tFold-TCR, and OpenFold3. Using EFT with TCR-XAI2, we demonstrate that (1) CNN and transformer models exhibit distinct learning trajectories; (2) TCR α and β evidence can conflict during learning, limiting the benefits of jointly modeling both chains, while MHC information mitigates this; and (3) real versus predicted structural data for TCR-epitope prediction exhibits distinct TCR and peptide feature preferences as well as differing trajectories of model certainty.

1 INTRODUCTION

Scientific applications across biology (Chen et al., 2024), medical image analysis (Shen et al., 2017), and neuroscience (Richards et al., 2019) have benefited from the effectiveness of deep learning (Jordan & Mitchell, 2015). However these models operate as black boxes, making their decision-making mechanisms difficult to understand and limiting their transparency and trustworthiness (Arrieta et al., 2020). Critically, lack of transparency also confounds researchers' ability to gather scientific insights and also to learn and diagnose model failures, and subsequently improve performance (Chen et al., 2024). Consequently, explainable artificial intelligence (XAI) has received significant attention in both academia and industry (Dwivedi et al., 2023). For example, Anthropic and OpenAI have investigated XAI to understand the internal mechanisms of AI models (Lindsey et al., 2025; Gao et al., 2025), while academia has developed methods for interpreting black-box models in computer vision (Chen et al., 2024), natural language processing (Zhao et al., 2024), and protein design (Hunklinger & Ferruz, 2026).

AI for science problems are particularly challenging because scientific data are often limited and the underlying biological mechanisms remain incompletely understood (Wang et al., 2023). This is particularly evident in T cell receptor (TCR)-epitope binding prediction, an important problem in adaptive immunity and a key step toward vaccine design and immunotherapy development (Joglekar & Li, 2021; Rojas et al., 2023; Poorebrahim et al., 2021). From a machine learning perspective, TCR-epitope binding prediction is a binary classification problem using up to four inputs: the TCR α chain, TCR β chain, peptide, and major histocompatibility complex (MHC) molecule, with each TCR chain containing three complementarity-determining regions (CDRs) (Hudson et al., 2023). Numerous models have been developed using diverse architectures, including transformers such as TULIP (Meynard-Piganeau et al., 2024) and MixTCRpred (Croce et al., 2024), and convolutional neural networks (CNNs) such as NetTCR-2.2 (Jensen & Nielsen, 2023). However, recent benchmarks, including IMMREP23 (Nielsen et al., 2024), IMMREP25 (Richardson et al., 2026), and Lu

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.

et al. (2026) benchmark, show that their generalization to unseen epitopes remains limited, while their black box nature makes it difficult to determine why and how they fail. Recent studies have addressed this challenge using post-hoc interpretation, such as QCAI for TULIP (Li et al., 2026a), and interpret-by-design approaches, such as TCR-SRIM (Li et al., 2026b) and TCR-EML (Li et al., 2025b). These methods provide insights into the evidence used by models and how they classify TCR-epitope pairs, and QCAI-guided model design has further demonstrated that such insights can improve prediction models (Li et al., 2025a). However, these approaches primarily interpret trained models and do not reveal why particular evidence is learned, ignored, or organized during training (Dwivedi et al., 2023). We therefore adopt a definition of explainability as the organization and evolution of interpretable evidence into a coherent structure, rather than the extraction of evidence from a trained black-box model after inference (Miller, 2019; LIPTON, 2018). Understanding this learning process is essential for explaining TCR-epitope binding prediction mechanisms. For example, recent work shows that an unpaired TCR chain can be sufficient for prediction (Shah et al., 2026). Post-hoc interpretability alone cannot explain why do the two chains provide conflicting information, highlighting the need to explain the learning process itself.

Our primary contribution this paper is to develop Explainability From Training (EFT), a new paradigm that tracks the evolution of model interpretations throughout training to provide structured model explanations. EFT is model-agnostic and can be applied to both interpret-by-design models (e.g., TCR-SRIM (Li et al., 2026b)), and black box models with post-hoc interpretations, including transformer-based TULIP (Meynard-Piganeau et al., 2024) and CNN-based NetTCR-2.2 (Jensen & Nielsen, 2023). Our second contribution is to develop TCR-XAI2, an extension of a crystal structure-based TCR-XAI benchmark, which contains 274 TCR-epitope structures (Li et al., 2026a); we extend TCR-XAI it by incorporating 144 additional experimentally resolved structures from TCR3D2.0 (Lin et al., 2025). TCR-XAI2 also includes structure predictions from AlphaFold3 (Abramson et al., 2024), tFold-TCR (Wu et al., 2025), Boltz-2 (Passaro et al., 2025), OpenFold3 (The OpenFold3 Team, 2025), and TCRModel2 (Yin et al., 2023), to allow a examination of the impact of experimental versus predicted structures in any particular TCR-epiope model that utilizes structure. We also annotate CDR and peptide residues according to their structural roles.

Our third contribution is the application of EFT and TCR-XAI2 to TCR-SRIM, TULIP, MixTCRpred, and NetTCR-2.2 to analyze their learning trajectories and characterize differences in how they learn TCR-epitope binding. Our studies reveal three findings: (1) CNN-based models rely more heavily on peptide information, whereas transformer-based models place greater emphasis on TCR information; (2) models without MHC input consistently exhibit opposing changes in interpretation quality between the TCR α and β chains, while MHC information reduces this effect, suggesting that models may struggle to effectively integrate paired TCR chains; and (3) there are significant differences in TCR and peptide feature preference and model certainty during training when using real versus predicted structures for model regularization.

2 Our Approach

We introduce our explainability from training (EFT) paradigm based on unified definitions of binary classification models and their corresponding interpretation methods. As shown in Fig. 1, EFT tracks changes in model interpretations throughout training to characterize their learning trajectories. We first track feature importance changes to obtain importance trajectories, then cluster features according to the similarity of their trajectories to obtain cluster-level interpretation trajectories. Finally, we organize these trajectories into model learning trajectories, providing explanations of why a model relies on specific features and how it organizes them as predictive evidences during training.

Predictive Models & Corresponding Interpretations. We first define binary classification models and their corresponding interpretations. Given an input $x \in \mathcal{X} = \mathbb{R}^N$, that is a N elements real number vector, a binary classification model produces a probability prediction $\hat{y} \in \hat{\mathcal{Y}} = [0, 1]$, and can be represented as $f : \mathcal{X} \to \hat{\mathcal{Y}},\ f \in \mathcal{F}$. A dataset is defined as

$$D = (x_i, y_i)_{i=1}^{N_\mathrm{D}}, \qquad x_i \in \mathcal{X}, \quad y_i \in \mathcal{Y} = \{0, 1\}, \tag{1}$$

where N_D denotes the number of input-label pairs. We define a training operator $\mathcal{T} : \mathcal{F} \times D \to \mathcal{F}$, which updates a model checkpoint using the training dataset. Starting from checkpoint f_t, the checkpoint after one training step is given by $f_{t+1} = \mathcal{T}(f_t, D)$.

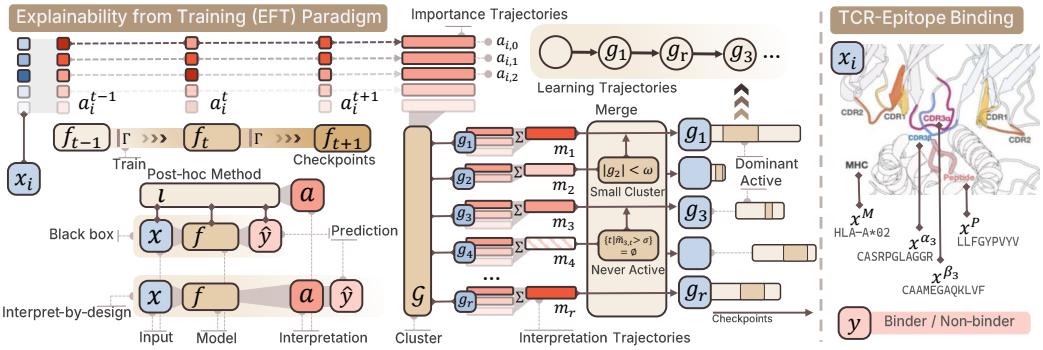

Figure 1: The EFT paradigm tracks model interpretations during training to obtain importance trajectories, clusters them into interpretation trajectories, and organizes them into learning trajectories that provide explanations. The diagram of TCR-epitope binding is from Parizi et al. (2026).

For a given checkpoint f_t, input x, and prediction $\hat{y} = f_t(x)$, its interpretation can be obtained in two ways. For black-box models, we use a post-hoc method ι to obtain feature importance scores:

$$\iota : \mathcal{F} \times \mathcal{X} \times \hat{\mathcal{Y}} \to \mathcal{A}, \qquad \mathcal{A} = [0,1]^N, \tag{2}$$

where $\mathcal{A}$ denotes the space of normalized feature importance scores with the same dimensionality as x. Different interpretation methods may produce different interaction structures. For example, GradCAM and AttnLRP assign a single importance score to each input feature, whereas QCAI can produce interaction-specific scores for features participating in different modalities. When multiple importance scores are assigned to the same feature, we average them to obtain a unified importance score. For interpret-by-design models, intermediate evidence is explicitly constructed as part of the prediction process. We therefore directly extract the corresponding feature importance scores from this evidence and transform them into the same representation $\mathcal{A}$.

Tracking Interpretations During Training. We first select a subset of samples as the tracking set, denoted by $D^{\mathrm{T}} \subseteq D$. Given the unified definitions of models and interpretation methods, we obtain the prediction of each checkpoint f_t for every sample in the tracking set as $f_t(x_i) = \hat{y}_i^t, (x_i, y_i) \in D^{\mathrm{T}}$. We then apply the corresponding interpretation method ι to obtain the importance

$$\iota(f_t, x_i, \hat{y}_i^t) = a_i^t \in \mathcal{A}^t, \ (x_i, y_i) \in D^{\mathrm{T}}, \tag{3}$$

where $\mathcal{A}^t$ is all importance scores of f_t on D^{T}. By repeating this process across all checkpoints, we obtain a trajectory of importance $\{\mathcal{A}^t\}_{t=0}^{N_T}$ for each tracking sample, which characterizes how the model's interpretation evolves throughout training.

Building Importance Trajectories. For each sample x_i and f_t, let a_i^t denote the corresponding vector of feature importance scores. For an input x_i, we represent the importance scores as

$$a_i^t = \left[a_{i,j}^t\right]_{j=0}^{N} = \left[a_{i,0}^t, a_{i,1}^t, \ldots, a_{i,N-1}^t\right], \tag{4}$$

where $a_{i,j}^t \in [0,1]$ denotes the importance of position j in input x_i at checkpoint t. For a fixed position j, its importance throughout training can therefore be represented as the importance trajectory

$$\mathbf{a}_{i,j} = \left[a_{i,j}^t\right]_{t=0}^{N_T} = \left[a_{i,j}^0, a_{i,j}^1, \ldots, a_{i,j}^{N_T}\right] \in [0,1]^{N_T}. \tag{5}$$

To identify when a feature becomes important during training, we define an activation threshold based on its importance trajectory. Given a $\tau \in (0,1]$ representing the top-τ fraction of checkpoints, we consider feature $x_{i,j}$ to be active at checkpoint t if its importance is within the top τ fraction of its importance values throughout training. The set of active checkpoints is therefore defined as

$$\pi_{i,j} = \left\{t \mid a_{i,j}^t \geq q_{1-\tau}\left(\mathbf{a}_{i,j}\right), \ t \in \{0, \ldots, N_T\}\right\}, \tag{6}$$

where $q_{1-\tau} : \mathbb{R}^{N_T} \to \mathbb{R}$ denotes the quantile function. Thus, $\pi_{i,j}$ identifies the checkpoints at which a feature exhibits relatively high importance compared with its own importance throughout training.

Feature Clustering and Interpretation Trajectories. Analyzing importance trajectories of individual features produce a large number of trajectories and can substantially reduce the readability of the resulting explanations. We therefore cluster features according to the consistency of importance trajectories throughout training to obtain interpretation trajectories. To make importance trajectories comparable across samples and positions, we normalize each of them across checkpoints: $\tilde{\mathbf{a}}_{i,j} = \mathbf{a}_{i,j} / \sum_{t=0}^{N_T} a_{i,j}^t$. We then apply K-means clustering to the normalized importance trajectories: $\mathcal{G}_K(\tilde{\mathbf{a}}) = \{g_r\}_{r=1}^K$, where $\mathcal{G}_K$ denotes K-means clustering with K clusters (interpretation trajectories). Each cluster g_r contains indices (i,j) corresponding to a specific sample and position. For each cluster, we summarize its importance at each checkpoint using the active features. Specifically, we define

$$m_{r,t} = \left(\sum\nolimits_{(i,j)\in g_{r,t}} a_{i,j}^t \right) / |g_{r,t}|, \qquad g_{r,t} = \{(i,j) \in g_r \mid t \in \pi_{i,j}\}, \tag{7}$$

where $g_{r,t}$ denotes the subset of features in cluster g_r that are active at checkpoint t. We then normalize the cluster importance across training checkpoints: $\tilde{m}_{r,t} = m_{r,t} / \max_{t'} m_{r,t'}$. A cluster is considered active at checkpoint t when $\tilde{m}_{r,t} > \sigma, \sigma \in (0,1]$. In addition to identifying active clusters, we seek to determine which evidence dominates the model's learning process at each checkpoint. We define cluster g_r as dominant at checkpoint t if it has the highest cluster importance and its importance exceeds a threshold ε:

$$\mathcal{U}_r^{\mathrm{d}} = \{t \mid r = \arg\max_{r'} m_{r',t}, \quad \max_{r'} m_{r',t} > \varepsilon\}. \tag{8}$$

Finally, we merge clusters that provide limited explanatory value to reduce the complexity of the resulting learning trajectories, also training-time explanations. A cluster g_r will be merged to nearest cluster g_m, if it is small: $|g_r| < \omega$, where $\omega \in \mathbb{Z}^+$, or if it is never active during training: $\{t \mid \tilde{m}_{r,t} > \sigma,\ t \in [0, N_{\mathrm{T}}]\} = \emptyset$. Then, this cluster g_r is merged into the nearest remaining cluster g_m, where

$$m = \arg\min_{r' \in [1,K]\setminus r} \sum\nolimits_{t=0}^{N_{\mathrm{T}}} \left| \left(\sum\nolimits_{(i,j)\in g_r} \tilde{a}_{i,j}^t \right) / |g_r| - \left(\sum\nolimits_{(i,j)\in g_{r'}} \tilde{a}_{i,j}^t \right) / |g_{r'}| \right|. \tag{9}$$

2.1 Applying EFT to TCR-epitope Prediction

TCR-epitope binding prediction can be formulated as a binary classification problem. We consider two types of inputs: amino acid sequences and MHC alleles. Amino acid sequences are represented as strings over the amino acid alphabet Θ (e.g., `CASSIVGGNEQFF`), while MHC alleles are represented as the allele set $\mathcal{M}$ (e.g., `HLA-A*02`). A TCR consists of an alpha and a beta chain, each containing three CDRs, which can be denoted as $x^{\mathrm{T}} = (x^c)_{c\in\mathcal{C}}$, $\mathcal{C} = \{\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3\}$, where $x^c \in \Theta^{N_{\mathrm{C}}}$ denotes the amino acid sequence of CDR region i. For simplicity, we assume that all CDRs are padded to a common length N_{C}. For example, CDR3a and CDR3b are represented by x^{α_3} and x^{β_3}, respectively. The peptide and MHC allele are represented as $x^{\mathrm{P}} \in \Theta^{N_{\mathrm{P}}}$ and $x^{\mathrm{M}} \in \mathcal{M}$, respectively, where N_{P} denotes the peptide length. A TCR-epitope sample is therefore represented as $x = x^{\mathrm{T}} \cup x^{\mathrm{P}} \cup x^{\mathrm{M}} \in \mathcal{X}$, where different models may use all or a subset of these inputs. Black box TCR-epitope models can be interpreted using standard post-hoc methods, whereas interpret-by-design models explicitly construct evidence for prediction. For example, TCR-SRIM produces contact prototypes that represent interactions between two input sequences before generating the final prediction. A contact prototype is represented as an interaction matrix $\phi \in \mathbb{R}^{N_{\mathrm{C}} \times N_{\mathrm{P}}}$, where $\phi^{\alpha\to\mathrm{P}}$ and $\phi^{\beta\to\mathrm{P}}$ denote the prototypes for CDR3a-peptide and CDR3b-peptide interactions, respectively. To obtain residue-level importance scores in the unified representation, we average each contact prototype along the corresponding dimension.

3 Results and Discussion

In this section, we first describe applying EFT to trace the training evolution of explanations for four TCR-epitope prediction models: NetTCR-2.2 (Jensen & Nielsen, 2023), MixTCRpred (Croce et al., 2024), TULIP (Meynard-Piganeau et al., 2024), and TCR-SRIM (Li et al., 2026b). NetTCR-2.2 is a CNN model, while the rest of models are transformer-based. Both NetTCR-2.2 and MixTCRpred concatenate all CDR regions as input, meanwhile, TULIP and TCR-SRIM using cross-attention to fuse only CDR3 and peptide inputs. TULIP provides two versions of model that with or with our

MHC allele as input. TCR-SRIM is the only interpret-by-design model that models CDR3-peptide interactions are contact maps, which allows to be trained with or without structure regularization. We first compare the version without structure regularization to align configurations of other models. Second, we discuss structure information in training using TCR-SRIM with structure regularization to compare the affects of experimental vs. predicted structures. We use the optimal interpretation method for each model: GradCAM (Selvaraju et al., 2017) for NetTCR-2.2, AttnLRP (Achtibat et al., 2024) for MixTCRpred, and QCAI (Li et al., 2026a) for TULIP. Each experiment is repeated three times with random seeds 0, 1, and 2. Ablation studies on interpretation methods and random seeds are provided in Appendix A.7 and Appendix A.6, respectively.

Next, we extend TCR-XAI benchmark (Li et al., 2026a) to build TCR-XAI2 dataset, provide an overview of its composition, and compare it with TCR-XAI. We quantitatively analyze these evolving explanations to investigate the potential conflicts between TCR α and β chains, the role of MHC alleles in model learning, and the TCR-epitope unbinding mechanism from the model's perspective. We further present case studies to illustrate these observations on specific examples. Finally, we compare TCR-SRIM models regularized with structures predicted by different structure prediction models to examine how structural information influences model learning. Based on these observations, we discuss implications for designing improved TCR-epitope prediction models.

3.1 TCR-XAI2 Benchmark

Li et al. (2026a) introduced TCR-XAI, a benchmark of 274 experimentally resolved TCR-epitope structures with residue-level contact maps as ground-truth explanations for quantitatively evaluating interpretation quality. To expand the benchmark, we collected the latest structures from the continuously updated TCR3D2.0 (Lin et al., 2025) and STCRDab (Leem et al., 2018), yielding 388 unique PDB structures. Since a single PDB entry may contain multiple copies of a TCR-epitope complex, we treated each copy as an independent sample, resulting in 619 samples. We further predicted the structures of these samples using AlphaFold3 (Abramson et al., 2024), Boltz-2 (Passaro et al., 2025), TCRModel2 (Yin et al., 2023), OpenFold3 (The OpenFold3 Team, 2025), and tFold-TCR (Wu et al., 2025). All methods generated predictions for the full set of samples except AlphaFold3 and TCRModel2, which failed for 15 and 28 samples, respectively. TCRModel2 additionally models only the TCR variable domains and peptide-binding groove, excluding β_2-microglobulin, the TCR constant domains, and the MHC α_3 domain. We combine these experimentally resolved and predicted structures to construct the TCR-XAI2 benchmark. We also provide training and test splits of TCR-XAI2 for models that require structural information during training, such as TCR-SRIM. To reduce sequence and structural leakage between the two splits, we evaluated the minimum Levenshtein distance between CDR3 and peptide sequences across the training and test sets. The median minimum distance is 18 residues, with a minimum of 5 residues, indicating substantial sequence differences between the splits. We further assessed structural differences using CDR3a-peptide and CDR3b-peptide contact maps, obtaining a median average contact-map distance of 2.65 Å.

Residue-level Structure Labels. We assign structural annotations to each residue in the CDR3a, CDR3b, and peptide regions of the TCR-XAI2 benchmark. These annotations describe each residue's structural role at the binding interface and are computed from either experimentally resolved or predicted structures. A contact label (e.g., `contact CDR3b`) is assigned when the closest heavy atom of a residue is within 5 Å of a residue in the corresponding partner region. A burial label (e.g., `buried CDR3b`) identifies a residue that buries more than 1 Å^2 of surface area from a partner upon binding. We calculate this value as the reduction in solvent-accessible surface area using the Lee-Richards algorithm with a probe radius of 1.40 Å. An geometric annotation (e.g., `geom+ CDR3b`) captures geometric rather than distance-based interactions and identifies residues that define the docking frame of the complex. Specifically, we use the conserved IMGT anchor positions 104 and 118 on each CDR3 loop and the N- and C-terminal residues of the peptide (Sanou et al., 2026). A residue may receive multiple labels because different structural roles can occur simultaneously, such as when a peptide residue contacts multiple CDR3 loops.

3.2 Model Explanations from Training

Using EFT, we first analyze the training-based explanations of four TCR-epitope binding prediction models: TCR-SRIM without structural regularization, TULIP without MHC allele as input,

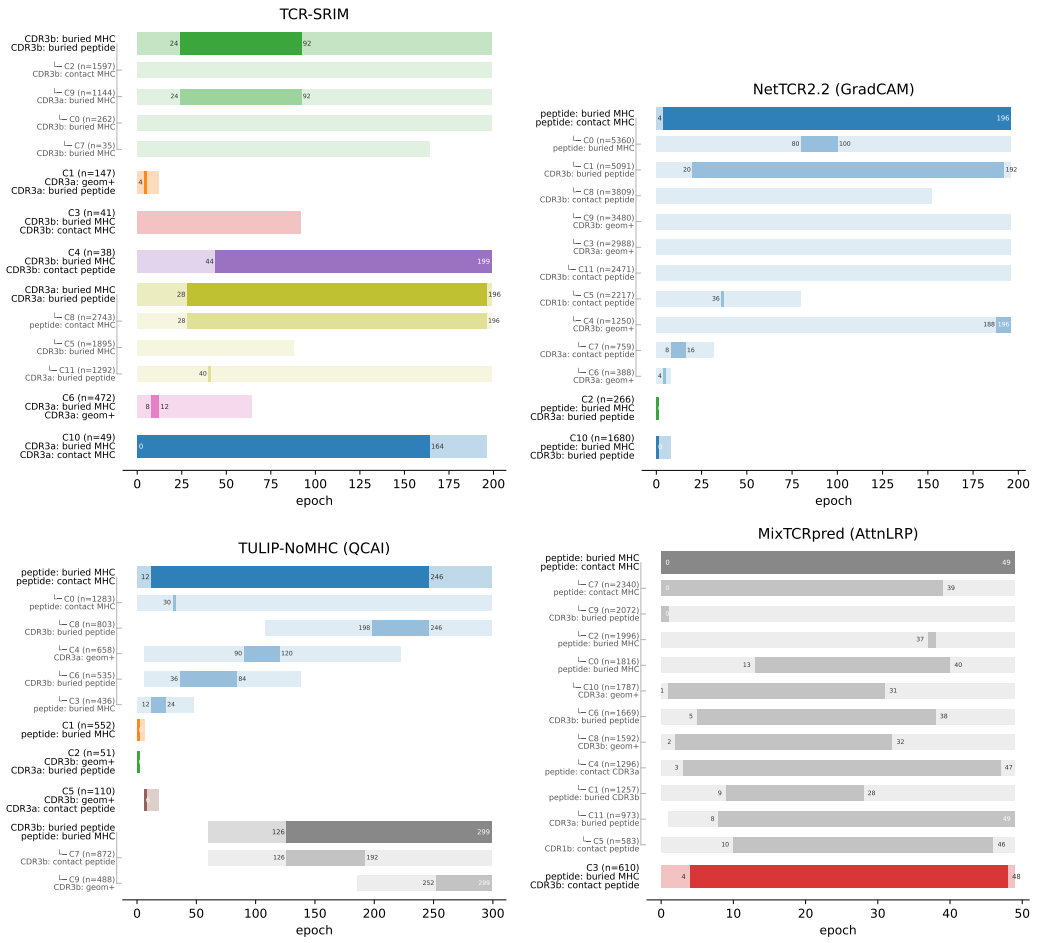

Figure 2: EFT explanations reveal distinct learning trajectories across CNN and transformer and feature fusion strategies, while showing that models struggle to learn TCR α and β simultaneously.

NetTCR-2.2, and MixTCRpred with peptide information as input. NetTCR-2.2 and MixTCRpred use all CDR regions as model inputs. As shown in Fig. 2, features are grouped into clusters according to their model representations. The light bars indicate features that are active during a given training period, while darker bars indicate clusters that dominate the model's explanation during that period. We use the structural labels defined from experimentally resolved structures to characterize each feature cluster. Because approximately 75% of peptide residues are buried within the MHC binding groove, the labels `peptide buried MHC` and `peptide contact MHC` dominate the training process of most models. TCR-SRIM is an exception because its contact prototype layers are explicitly designed to capture TCR-peptide interactions. To provide more informative descriptions, we report the top two or three structural labels for each cluster. We first characterize each cluster using its top two labels and group clusters with the same top two labels into a super-cluster. We then use the third-ranked label to distinguish and characterize the child clusters within each super-cluster.

The models primarily rely on CDR3 regions and peptides for TCR-epitope binding prediction. Even for NetTCR-2.2 and MixTCRpred, only one feature cluster is associated with CDR1b-peptide contact. Although both models use the same inputs, NetTCR-2.2 primarily focuses on CDR3b burial of the peptide, which dominates across most training epochs, whereas MixTCRpred emphasizes CDR3a and exhibits more diverse dominant feature clusters. These differences suggest that the CNN-based NetTCR-2.2 and transformer-based MixTCRpred organize TCR-epitope evidence differently, with MixTCRpred distributing evidence across a broader range of interaction features.

Beyond these observations, we identify two additional patterns in the training process. For models other than TCR-SRIM, the models consistently rely on peptide-MHC interactions throughout training, particularly during the early and late stages. In contrast, CDR-peptide and CDR-MHC

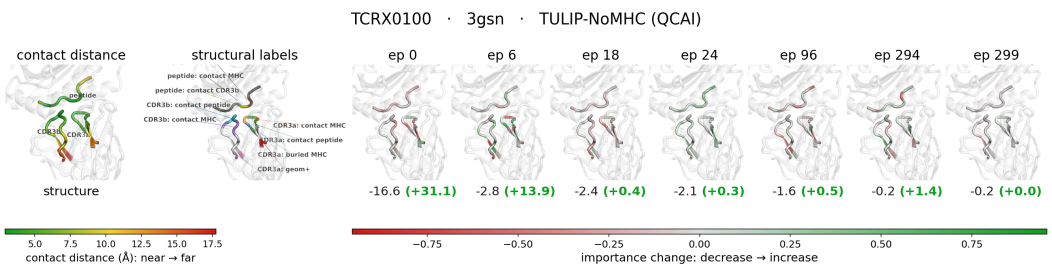

Figure 3: A EFT case study of TULIP-NoMHC on a human cytomegalovirus sample reveals conflicts between TCR α and β information for model learning during training.

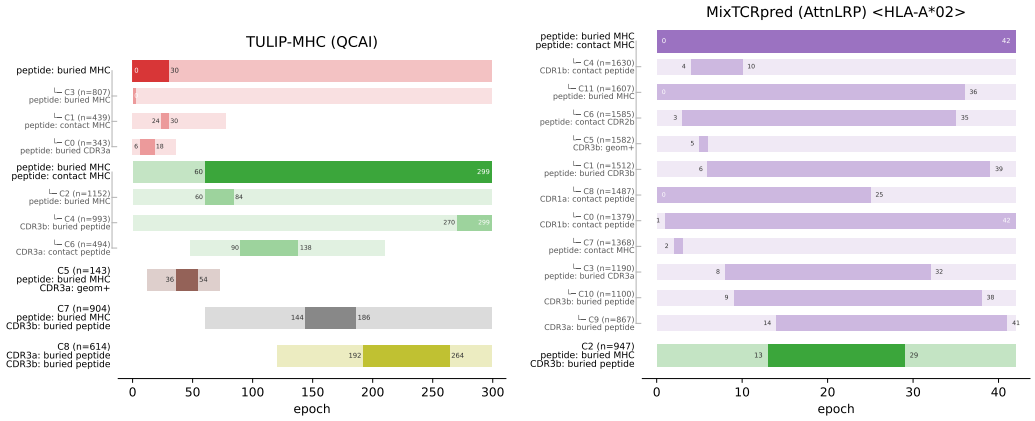

Figure 4: EFT explanations for TULIP-MHC and MixTCRpred trained on samples with only the HLA−A⋆02 allele show that MHC information can reduce conflicts between TCR α and β information and enable models to learn features from more chains.

interactions primarily emerge during intermediate stages of training. Furthermore, the models tend to learn CDR3a-related information before CDR3b-related information, and do not learn TCR alpha and beta information simultaneously.

We then present a case study using EFT to analyze TULIP without MHC allele information to demonstrate EFT explanation for a single case. It corresponds to a human cytomegalovirus infection sample from PDB structure 3GSN. Consistent with the general explanation analysis, TULIP initially focuses on the CDR3a-peptide interaction and progressively increases its attention to peptide residues involved in CDR3a contacts during the early training stages. We observe a clear contrast between the alpha and beta chains from epochs 0 to 18. The residue-level attention patterns shift in opposite directions between the two chains, indicating an alpha-beta conflict in the evidence used by the model. This can explain why model does not learn TCR α and β information simultaneously.

3.2.1 Role of MHC and TCR Chain Conflicts in Model Learning

To further investigate why models cannot learn TCR alpha and beta information simultaneously and how MHC alleles can influence TCR-epitope model learning, we analyze EFT explanations for TULIP with MHC allele information as an input and for MixTCRpred trained exclusively on samples with the HLA−A⋆02 allele. As shown in Fig. 4, providing MHC allele information or restricting training data to a specific MHC allele leads to distinct learning trajectories. For TULIP with MHC information, a feature cluster corresponding to CDR3a and CDR3b burial of the peptide remains dominant for more than 50 epochs starting from epoch 192. In contrast, for the model without MHC information, the CDR3a-peptide and CDR3b-peptide interaction features never become dominant simultaneously. The difference is more pronounced for MixTCRpred trained exclusively on HLA−A⋆02 samples, where CDR1a and CDR2b also emerge as important evidence. These observations suggest that MHC information can play two complementary roles in TCR-epitope model

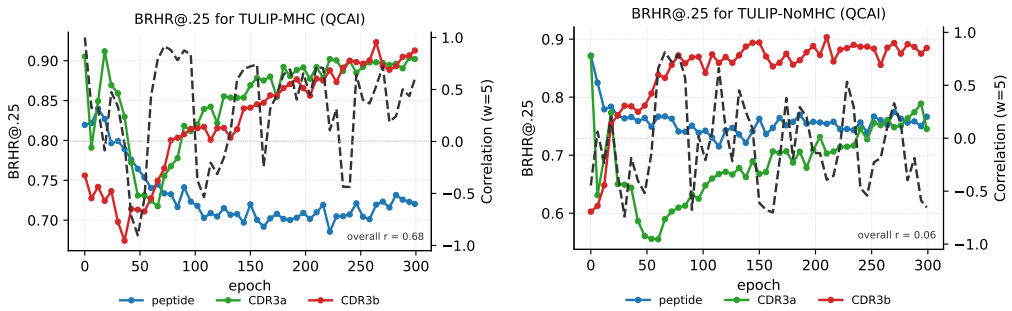

Figure 5: The evolution of BRHR@.25 for CDR3a, CDR3b, and peptide during TULIP training with and without MHC allele information quantitatively demonstrates that MHC information can reduce conflicts between TCR α and β information.

learning. When MHC information is not provided and the training data cover diverse MHC alleles, the model may need to infer MHC-dependent information indirectly from the TCR and peptide sequences. This can lead the model to associate peptide recognition with one TCR chain as a proxy for MHC-dependent effects. In contrast, when MHC information is provided or the training data are restricted to a specific MHC allele, the model can allocate more capacity to other input features and jointly integrate information from the alpha and beta chains.

In other hand, this learning process may create a conflict between the two TCR chains when MHC information is unavailable. Before learning to coordinate both chains, the model may rely more heavily on one chain to recognize the peptide, limiting its use of complementary information from the other chain. This provides a potential explanation for why incorporating paired TCR chains does not necessarily yield substantially higher predictive performance than using unpaired chains (Shah et al., 2026). This is also the reason why Li et al. (2025a) finds adding a cross-attention to fusion TCR alpha and beta chains information can improve both interpretation quality and performance.

3.3 Interpretability During Training

To further examine the roles of MHC information and potential conflicts between the alpha and beta chains during model learning, we quantitatively track binding region hit rate (BRHR) throughout training for the peptide, CDR3a, and CDR3b regions, which is introduced by Li et al. (2026a). We compare TULIP trained with and without MHC allele information. For CDR3a and CDR3b, we measure BRHR@.25 for the CDR3a-peptide and CDR3b-peptide interactions, respectively. For the peptide, we use the average BRHR@.25 across its interactions with CDR3a and CDR3b.

As shown in Fig. 5, the evolution of BRHR during training further supports our previous observations regarding the role of MHC information and potential conflicts between the TCR alpha and beta chains. In addition to tracking the BRHR of CDR3a and CDR3b separately, we compute their sliding-window Pearson correlation using a window size of 5. When MHC allele information is provided as an input, the BRHR trajectories of CDR3a and CDR3b are substantially more consistent, with an overall Pearson correlation of $r = 0.68$. In contrast, without MHC information, the BRHR difference between CDR3a and CDR3b exceeds 0.1, while their overall correlation is only $r \approx 0.06$. Notably, strong fluctuations and opposing trends are observed during the early stages of training. These results further support our previous observation that providing MHC allele information reduce the effort required to infer the peptide-MHC relationship indirectly, allowing the model to allocate more capacity to coordinating information from the alpha and beta chains. The consistent BRHR trajectories observed when MHC information is available suggest that the model can learn more coordinated evidence from the two TCR chains.

3.4 Experimental versus Predicted Structure for Guiding Model Training

Recent studies have explored incorporating structural information into TCR-epitope prediction models (Deleuran & Nielsen, 2025; Li et al., 2026b). Motivated by these approaches, we use TCR-XAI2

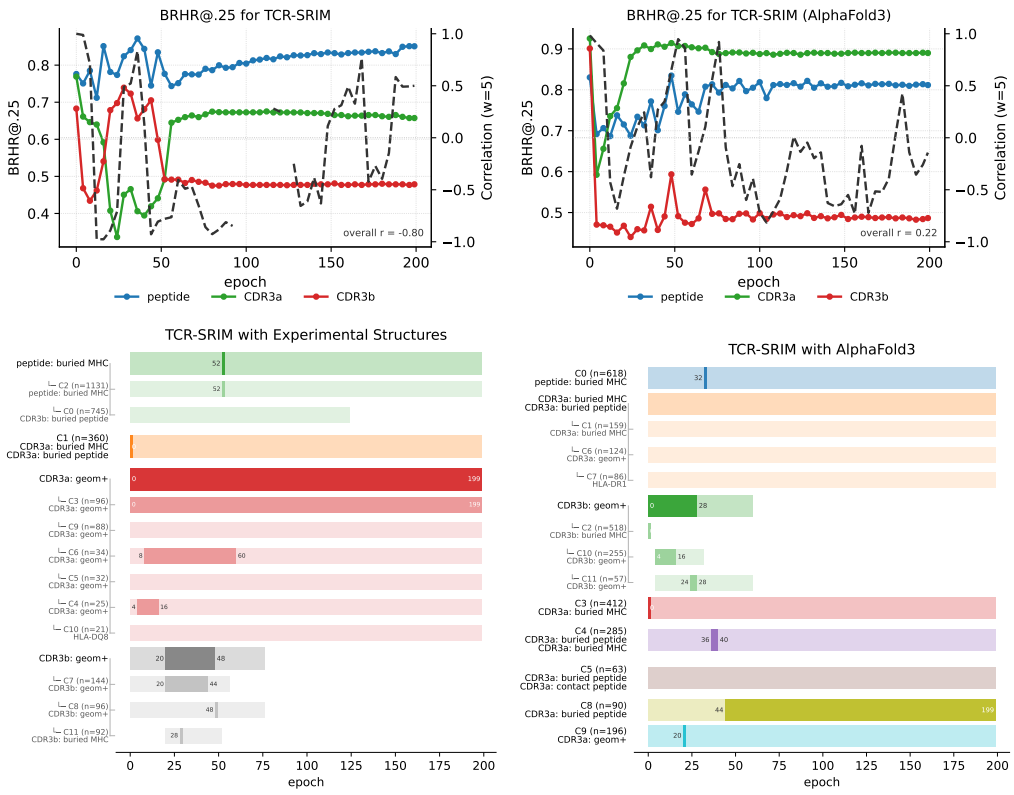

Figure 6: EFT explanations and evolution of BRHR@.25 during training for TCR-SRIM regularized with experimentally resolved structures and AlphaFold3-predicted structures. Structural information rapidly stabilizes model interpretations, while experimentally resolved and predicted structures lead to different interpretation trajectories during training.

structures predicted by different structure prediction models to investigate how structural quality affects model learning. As a baseline, we first examine EFT explanations and BRHR evolution in TCR-SRIM without structural regularization. The BRHR trajectories of all three chains fluctuate substantially throughout training rather than converging to stable patterns, indicating that structural evidence remains unstable without structural guidance. We then regularize TCR-SRIM using experimentally resolved or AlphaFold3-predicted structures from the TCR-XAI2 training split and evaluate BRHR on the test split using experimentally resolved structures. As shown in Fig. 6, experimentally resolved structures lead to stable BRHR values for all three chains within approximately 70 epochs, with substantially reduced fluctuations. AlphaFold3-predicted structures also stabilize model interpretations. However, EFT shows that both models primarily focus on CDR3a and CDR3a-peptide interactions, while CDR3b-peptide interactions receive greater attention mainly during early training, consistent with the BRHR analysis. This behavior result from the absence of MHC information, leading the model to rely more heavily on a single TCR chain for prediction.

4 Conclusion

In this paper, we proposed the Explainability From Training (EFT) paradigm. For a given model and interpretability method, the output of EFT is a trajectory that captures tracks how the model learns and uses features. To demonstrate its utility, we apply EFT to four state-of-the-art TCR-epitope models. With the updated TCR-XAI2 benchmark, we analyzed these models and demonstrate that: (1) CNNs rely more on peptide information, while transformers emphasize TCR features; (2) models without MHC input show conflicting TCR α- and β-chain evidence, which is reduced by MHC information; and (3) structural differences between experimentally resolved and predicted structures affect models' preference for TCR or peptide information and trajectories of model certainty.

AI USE STATEMENT

In this work, we used generative AI tools for implement methods. We have not used generative AI tools for Generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, clean and reformat dataset, support qualitative and thematic data analysis, interpret results, and assist with translation are not applicable to this work. Additionally, we used generative AI tools for modify scientific figures or images, create or edit software code, sourcing/searching for information, edit a research paper to improve readability, propose a title or keywords for a research paper. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

For the review stage, we provide a ZIP file containing the code and instructions for reviewers to reproduce and verify our results. We will release the code in a public repository and publish the trained model weights upon publication.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pp. 1–3, 2024.

Reduan Achtibat, Sayed Mohammad Vakilzadeh Hatefi, Maximilian Dreyer, Aakriti Jain, Thomas Wiegand, Sebastian Lapuschkin, and Wojciech Samek. Attnlrp: Attention-aware layer-wise relevance propagation for transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

Rémy Bosselut. T cell antigen recognition: Evolution-driven affinities. *Proceedings of the National Academy of Sciences*, 116(44):21969–21971, 2019.

Valerie Chen, Muyu Yang, Wenbo Cui, Joon Sik Kim, Ameet Talwalkar, and Jian Ma. Applying interpretable machine learning in computational biology—pitfalls, recommendations and opportunities for new developments. *Nature methods*, 21(8):1454–1461, 2024.

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.

Mark M Davis and Pamela J Bjorkman. T-cell antigen receptor genes and t-cell recognition. *Nature*, 334(6181):395–402, 1988.

Sebastian Nymann Deleuran and Morten Nielsen. Nettcr-struc, a structure driven approach for prediction of tcr-pmhc interactions. *Frontiers in Immunology*, 16:1616328, 2025.

Rudresh Dwivedi, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, et al. Explainable ai (xai): Core ideas, techniques, and solutions. *ACM computing surveys*, 55(9):1–33, 2023.

Leo Gao, Achyuta Rajaram, Jacob Coxon, Soham V Govande, Bowen Baker, and Dan Mossing. Weight-sparse transformers have interpretable circuits. *arXiv preprint arXiv:2511.13653*, 2025.

Dan Hudson, Ricardo A Fernandes, Mark Basham, Graham Ogg, and Hashem Koohy. Can we predict t cell specificity with digital biology and machine learning? *Nature Reviews Immunology*, 23(8):511–521, 2023.

Dan Hudson, Alex Lubbock, Mark Basham, and Hashem Koohy. A comparison of clustering models for inference of t cell receptor antigen specificity. *ImmunoInformatics*, 13:100033, 2024.

Andrea Hunklinger and Noelia Ferruz. Towards the explainability of protein language models. *Nature Machine Intelligence*, pp. 1–14, 2026.

IMMREP25. Immrep25: Tcr specificity prediction challenge. `https://kaggle.com/competitions/immrep25`, 2025. Kaggle.

Mathias Fynbo Jensen and Morten Nielsen. Nettcr 2.2-improved tcr specificity predictions by combining pan-and peptide-specific training strategies, loss-scaling and integration of sequence similarity. *bioRxiv*, pp. 2023–10, 2023.

Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Layercam: Exploring hierarchical class activation maps for localization. *IEEE transactions on image processing*, 30:5875–5888, 2021.

Alok V Joglekar and Guideng Li. T cell antigen discovery. *Nature methods*, 18(8):873–880, 2021.

Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

I Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle Friedler. Problems with shapley-value-based explanations as feature importance measures. In *International conference on machine learning*, pp. 5491–5500. PMLR, 2020.

Jinwoo Leem, Saulo H P de Oliveira, Konrad Krawczyk, and Charlotte M Deane. Stcrdab: the structural t-cell receptor database. *Nucleic acids research*, 46(D1):D406–D412, 2018.

Jiarui Li, Zixiang Yin, Zhengming Ding, Samuel J Landry, and Ramgopal R Mettu. Rational multimodal transformers for tcr-pmhc prediction. In *Proceedings of the 16th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pp. 1–10, 2025a.

Jiarui Li, Zixiang Yin, Zhengming Ding, Samuel J Landry, and Ramgopal R Mettu. Tcr-eml: Explainable model layers for tcr-pmhc prediction. *arXiv preprint arXiv:2510.04377*, 2025b.

Jiarui Li, Zixiang Yin, Haley Smith, Zhengming Ding, Samuel J Landry, and Ramgopal R. Mettu. Quantifying cross-attention interaction in transformers for interpreting TCR-pMHC binding. In *The Fourteenth International Conference on Learning Representations*, 2026a. URL `https://openreview.net/forum?id=S3kSOFhs5m`.

Jiarui Li, Zixiang Yin, Yunbei Zhang, Janet Wang, Samuel J Landry, Zhengming Ding, and Ramgopal R Mettu. Structure-regularized interpretable tcr-epitope prediction. *arXiv preprint arXiv:2606.30902*, 2026b.

Valerie Lin, Melyssa Cheung, Ragul Gowthaman, Maya Eisenberg, Brian M Baker, and Brian G Pierce. Tcr3d 2.0: expanding the t cell receptor structure database with new structures, tools and interactions. *Nucleic Acids Research*, 53(D1):D604–D608, 2025.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson. On the biology of a large language model. Technical report, Anthropic, 2025. `https://transformer-circuits.pub/2025/attribution-graphs/biology.html` (Accessed: 2026-03-02).

ZACHARY C LIPTON. The mythos of model interpretability. *COMMUNICATIONS OF THE ACM*, 61(10), 2018.

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.

Koshlan Mayer-Blackwell, Stefan Schattgen, Liel Cohen-Lavi, Jeremy C Crawford, Aisha Souquette, Jessica A Gaevert, Tomer Hertz, Paul G Thomas, Philip Bradley, and Andrew Fiore-Gartland. Tcr meta-clonotypes for biomarker discovery with tcrdist3 enabled identification of public, hla-restricted clusters of sars-cov-2 tcrs. *Elife*, 10:e68605, 2021.

Barthelemy Meynard-Piganeau, Christoph Feinauer, Martin Weigt, Aleksandra M Walczak, and Thierry Mora. Tulip: A transformer-based unsupervised language model for interacting peptides and t cell receptors that generalizes to unseen epitopes. *Proceedings of the National Academy of Sciences*, 121(24):e2316401121, 2024.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

Jacques Neefjes, Marlieke LM Jongsma, Petra Paul, and Oddmund Bakke. Towards a systems understanding of mhc class i and mhc class ii antigen presentation. *Nature reviews immunology*, 11(12):823–836, 2011.

Morten Nielsen, Anne Eugster, Mathias Fynbo Jensen, Manisha Goel, Andreas Tiffeau-Mayer, Aurelien Pelissier, Sebastiaan Valkiers, María Rodríguez Martínez, Barthélémy Meynard-Piganeeau, Victor Greiff, et al. Lessons learned from the immrep23 tcr-epitope prediction challenge. *ImmunoInformatics*, 16:100045, 2024.

Farzaneh M Parizi, Yannick JM Aarts, Nils Smit, AR Dona Roran, Daniëlle Diepenbroek, Wieke A Krösschell, Levin Thijs, Joost Tepperik, Dario F Marzella, Gayatri Ramakrishnan, et al. Swifttcr: Efficient computational docking protocol of tcrpmhc-i complexes using restricted rotation matrices. *Briefings in Bioinformatics*, 27(3):bbag306, 2026.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Mansour Poorebrahim, Niloufar Mohammadkhani, Reza Mahmoudi, Monireh Gholizadeh, Elham Fakhr, and Angel Cid-Arregui. Tcr-like cars and tcr-cars targeting neoepitopes: an emerging potential. *Cancer gene therapy*, 28(6):581–589, 2021.

Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al. A deep learning framework for neuroscience. *Nature neuroscience*, 22(11):1761–1770, 2019.

Eve Richardson, Yannick Jurriaan Maria Aarts, John A Altin, Coos AB Baakman, Philip Bradley, Binbin Chen, Joakim Clifford, Manjima Dhar, Danielle Diepenbroek, Ethan Fast, et al. Immrep25: Unseen peptides. *bioRxiv*, pp. 2026–03, 2026.

Luis A Rojas, Zachary Sethna, Kevin C Soares, Cristina Olcese, Nan Pang, Erin Patterson, Jayon Lihm, Nicholas Ceglia, Pablo Guasp, Alexander Chu, et al. Personalized rna neoantigen vaccines stimulate t cells in pancreatic cancer. *Nature*, 618(7963):144–150, 2023.

Gaoussou Sanou, Guilhem Zeitoun, Taciana Manso, Milad Eidi, Shamsa Batool, Anjana Kushwaha, François Grand, Myriam Croze, Axel Vaillant, Chahrazed Debbagh, et al. Imgt® at scale: Fair, dynamic, and automated tools for immune locus analysis. *Nucleic Acids Research*, 54(D1): D1119–D1132, 2026.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Aisha Shah, Raphael Genolet, Aymeric Auger, Dana Léa Moreno, Yan Liu, Giancarlo Croce, Julien Racle, Alexandre Harari, and David Gfeller. Unpaired tcrα+ tcrβ sequencing is sufficient for training machine learning tcr-epitope recognition predictors. *bioRxiv*, pp. 2026–03, 2026.

Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual review of biomedical engineering*, 19:221–248, 2017.

The OpenFold3 Team. Openfold3-preview, 2025. URL https://github.com/aqlaboratory/openfold-3.

Sebastiaan Valkiers, Max Van Houcke, Kris Laukens, and Pieter Meysman. Clustcr: a python interface for rapid clustering of large sets of cdr3 sequences with unknown antigen specificity. *Bioinformatics*, 37(24):4865–4867, 2021.

Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, et al. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023.

Fandi Wu, Yu Zhao, Yang Xiao, Chenchen Qin, Fang Wang, Zihan Wu, Long-Kai Huang, Xiao Liu, Jiangning Song, Bing He, et al. Fast and accurate modeling of tcr-peptide-mhc complexes using tfold-tcr. *bioRxiv*, pp. 2025–01, 2025.

Rui Yin, Helder V Ribeiro-Filho, Valerie Lin, Ragul Gowthaman, Melyssa Cheung, and Brian G Pierce. Tcrmodel2: high-resolution modeling of t cell receptor recognition using deep learning. *Nucleic Acids Research*, 51(W1):W569–W576, 2023.

Hongyi Zhang, Xiaowei Zhan, and Bo Li. Giana allows computationally-efficient tcr clustering and multi-disease repertoire classification by isometric transformation. *Nature communications*, 12 (1):4699, 2021.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38, 2024.

A APPENDIX

A.1 TCR-EPITOPE BINDING INTRODUCTION

TCR-epitope recognition is a key step in T cell activation and adaptive immune responses. Antigens are first taken up by antigen-presenting cells (APCs) and processed into peptides. These peptides bind to MHC molecules to form peptide-MHC (pMHC) complexes, which are presented on the APC surface and recognized by $\alpha\beta$-TCRs on T cells (Davis & Bjorkman, 1988; Neefjes et al., 2011). MHC molecules are generally classified into MHC-I and MHC-II, which primarily present intracellular antigens recognized by CD4+ T cell, such as tumor-associated antigens, and exogenous antigens recognized by CD8 T cell, such as viral antigens, respectively (Hudson et al., 2023). An $\alpha\beta$-TCR consists of one alpha chain and one beta chain, each containing three complementarity-determining regions (CDRs). These regions, particularly CDR3, play important roles in determining whether a TCR recognizes a pMHC complex. The six CDRs are CDR1a, CDR2a, CDR3a, CDR1b, CDR2b, and CDR3b (Bosselut, 2019).

A.2 TCR-EPITOPE PREDICTION MODELS.

TCR-epitope prediction models can be broadly categorized into unsupervised and supervised approaches (Hudson et al., 2024). Unsupervised approaches primarily rely on sequence similarity and clustering, with representative methods including TCRdist3 (Mayer-Blackwell et al., 2021), GIANA (Zhang et al., 2021), and ClusTCR (Valkiers et al., 2021). Recent advances have increasingly shifted toward supervised learning, with models adopting diverse neural architectures. These approaches include CNNs, such as NetTCR-2.2 (Jensen & Nielsen, 2023), and transformer-based models, such as TULIP (Meynard-Piganeau et al., 2024), MixTCRpred (Croce et al., 2024), and TCR-SRIM (Li et al., 2026b). NetTCR-2.2 and MixTCRpred concatenate CDR regions with the peptide sequence for binding prediction. TULIP uses CDR3 regions and peptide sequences, with an optional MHC allele as input. TCR-SRIM uses CDR3 regions and peptide sequences to predict TCR-epitope binding.

A.3 MODEL INTERPRETABILITY.

Deep learning models can be broadly categorized into black-box and interpret-by-design models. Interpret-by-design models, such as TCR-SRIM, incorporate interpretable components directly into their architectures. In contrast, black-box models can be analyzed using post-hoc interpretability methods, which estimate the contribution of input features to model predictions. These methods can be broadly categorized into model-agnostic and model-specific approaches. SHAP is a representative model-agnostic method that estimates feature contributions based on Shapley values (Kumar et al., 2020). Model-specific methods are designed to exploit the architectures or internal representations of particular model families. For CNNs, class activation map (CAM) methods, including Grad-CAM (Selvaraju et al., 2017) and LayerCAM (Jiang et al., 2021), are commonly used to identify spatially important features. For transformer-based models, layer-wise relevance propagation (LRP), such as AttnLRP (Achtibat et al., 2024), and attention-based methods, such as QCAI (Li et al., 2026a), have been used to analyze the contribution of individual tokens or interactions to model predictions.

A.4 TCR-XAI2 BENCHMARK

Li et al. (2026a) introduced the TCR-XAI benchmark, which was compiled from experimentally resolved crystal structures to quantitatively evaluate interpretation quality. It contains 274 samples and uses residue-level contact maps as ground-truth explanations. Since TCR3D2.0 (Lin et al., 2025) is continuously updated with newly resolved TCR-epitope structures, we collected the latest structures from TCR3D2.0 together with structures from STCRDab (Leem et al., 2018) to construct the TCR-XAI2 benchmark. This process yielded 388 unique structures. Because a single PDB entry may contain multiple copies of the same TCR-epitope complex, we treated each copy as an independent sample to increase the number of samples of the benchmark, resulting in 619 samples in total. Duplicate copies are shown in Fig. 7.

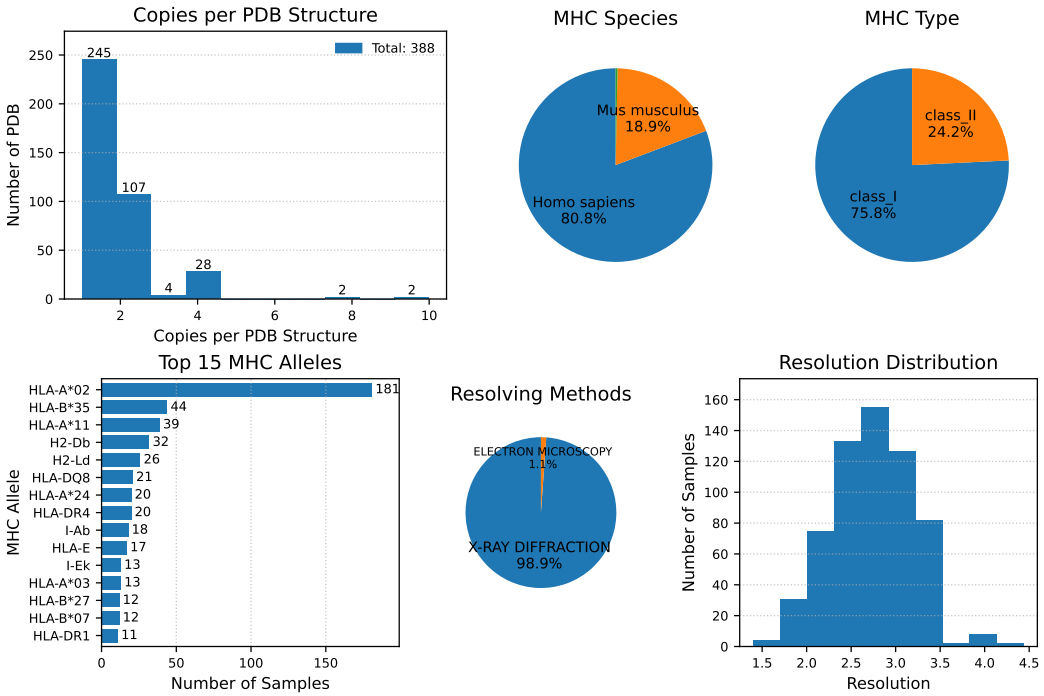

Figure 7: **The statistics of sample distribution of TCR-XAI2 benchmark.** The TCR-XAI2 benchmark consists of 388 unique structures and 619 samples, while a structure can provide multiple TCR-epitope copies. Most of the samples are from human (80.8%), MHC-I (75.8%), and resolved by X-ray diffraction (98.9%).

Among the TCR-XAI2 samples, 80.8% contain human MHC molecules, 18.9% contain mouse MHC molecules, and 0.3% contain MHC molecules from other species. Of these samples, 75.8% involve MHC-I and 24.2% involve MHC-II. The distribution of MHC alleles is highly imbalanced. The three most frequent human alleles are `HLA-A*02` (181 samples), `HLA-B*35` (44 samples), and `HLA-A*11` (39 samples), consistent with the allele distribution commonly observed in sequence-based TCR-epitope datasets. The two most frequent mouse alleles are `H2-Db` (32 samples) and `H2-Ld` (26 samples). Most samples (98.9%) were resolved using X-ray diffraction, while the remaining samples were resolved using cryo-electron microscopy (cryo-EM). The structures have a mean resolution in the range of 2.5 to 3.0 Å.

We also provide training and test splits of TCR-XAI2 for models such as TCR-SRIM, which include variants that require structural information during training. To minimize potential data leakage, we designed the split to account for differences in both sequence and structure, specifically the CDR3b-peptide and CDR3a-peptide contact maps. As shown in Fig. 8, we evaluated the minimum Levenshtein distance of CDR3 regions and peptides between the training and test splits. The median distance is 18 residues, with a minimum of 5 residues, indicating substantial sequence differences between the two splits. We further evaluated structural differences using the CDR3a-peptide and CDR3b-peptide contact maps. The median average contact distance is 2.65 Å. These results indicate that the training and test splits are separated in both sequence and structural space, reducing the potential for data leakage from either source.

Predicted Structures. To further investigate how predicted structures influence model explanations and training, we used AlphaFold3 (Abramson et al., 2024), Boltz-2 (Passaro et al., 2025), TCRModel2 (Yin et al., 2023), OpenFold3 (The OpenFold3 Team, 2025), and tFold-TCR (Wu et al., 2025) to predict the structures corresponding to the TCR-XAI2 benchmark. For AlphaFold3 and TCRModel2, we used their official web servers to generate predictions. We deployed tFold-TCR locally, while Boltz-2 and OpenFold3 predictions were generated through the NVIDIA NIM API. As shown in Table 1, we divided each predicted-structure dataset into training and test splits using the

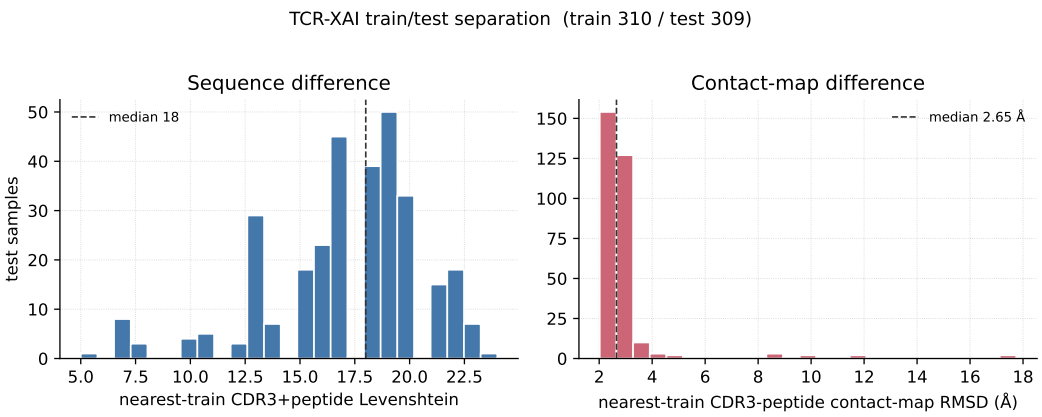

Figure 8: **The sequence and structural differences between the training and test splits of the TCR-XAI2 benchmark.** From the sequence perspective, the median distance between the training and test splits is 18 residues, with a minimum distance of 5 residues. From the structural perspective, the CDR3-peptide contact maps exhibit a median difference of at least 2.65 Åbetween the training and test splits.

same splitting criteria as the experimentally resolved structures. Except for AlphaFold3 and TCRModel2, all methods successfully generated structures for the full set of 619 samples. AlphaFold3 and TCRModel2 failed to generate predictions for 15 and 28 samples, respectively. In addition, TCRModel2 models only the TCR variable domains and the peptide-binding groove, omitting β_2-microglobulin, the TCR constant domains, and the MHC α_3 domain.

Table 1: **The number of samples and training and test splits in the predicted TCR-XAI2 benchmarks.** AlphaFold3 and TCRModel2 failed to predict 15 and 28 samples, respectively, while all other methods successfully predicted structures for all samples.

Model	Train	Test	Total	Missing
TCR-XAI2 (AlphaFold3)	303	301	604	15
TCR-XAI2 (Boltz-2)	310	309	619	0
TCR-XAI2 (OpenFold)	310	309	619	0
TCR-XAI2 (TCRModel2)	298	293	591	28
TCR-XAI2 (tFold-TCR)	310	309	619	0
TCR-XAI2	310	309	619	-

We further evaluated the quality of the predicted structures using full-structure RMSD and contact-map RMSD for the CDR3a-peptide and CDR3b-peptide interfaces, with the experimentally resolved structures as references. As shown in Table 1, TCRModel2 is not directly comparable in terms of full-structure RMSD because it does not provide the complete complex structure. Boltz-2 achieves the lowest RMSD among the evaluated methods, with CDR3-peptide interface RMSDs below 2.5 Åand full-structure RMSDs below 3.5 Å. AlphaFold3 and TCRModel2 also achieve CDR3-peptide contact-map RMSDs below 3.5 Å, suggesting that their predicted interfaces remain reasonably close to the experimentally resolved structures. In contrast, tFold-TCR and OpenFold3 show substantially larger CDR3-peptide contact-map RMSDs, exceeding 6 Åand 15 Å, respectively.

Proxy Resolution for Predicted Structures. To estimate a proxy resolution for predicted structures, we compute a score from the predicted confidence of the CDR3a, CDR3b, and peptide regions following (Li et al., 2026b). AlphaFold3, Boltz-2, OpenFold3, and TCRModel2 provide per-residue pLDDT scores $\phi_i \in [0, 100]$. For each region $c \in \text{CDR3}a, \text{CDR3}b, \text{Peptide}$, we first compute its mean pLDDT:

$$\bar{\phi}_c = \frac{1}{|c|} \sum_{i \in c} \phi_i. \tag{10}$$

Table 2: **The structural quality of predictions from different models.** We compare the mean RMSD and standard deviation of the full structure and CDR3-peptide interfaces between predicted and experimentally resolved structures. Full-structure RMSD is not available for TCRModel2 because it omits several components of the complete TCR-pMHC complex.

Predictor	CDR3a-peptide (Å)	CDR3b-peptide (Å)	Full (Å)
Boltz-2	$\mathbf{2.48 \pm 7.78}$	$\mathbf{2.38 \pm 7.59}$	$\mathbf{3.24 \pm 5.88}$
AlphaFold3	2.97 ± 7.88	2.95 ± 7.74	3.75 ± 6.01
TCRmodel2	3.38 ± 7.98	3.35 ± 7.74	-
tFold-TCR	6.35 ± 7.54	7.28 ± 6.89	10.29 ± 7.48
OpenFold3	23.47 ± 13.56	16.47 ± 8.14	35.44 ± 3.30

We then use the minimum regional confidence as the overall confidence of the complex:

$$\phi_{\min} = \min\left(\bar{\phi} * \mathrm{CDR3}\alpha, \bar{\phi} * \mathrm{CDR3}\beta, \bar{\phi} * \mathrm{Peptide}\right). \tag{11}$$

Finally, we convert this confidence score to a proxy resolution:

$$r = \frac{100 - \phi * \min}{10}. \tag{12}$$

For tFold-TCR, whose pLDDT scores are normalized to $[0, 1]$, we instead compute

$$r = 10(1 - \phi_{\min}). \tag{13}$$

Thus, lower r indicates higher predicted structural confidence.

Binding Region Hit Rate. To quantitatively assess the quality of model interpretations and attention-based importance scores, we adopt the binding region hit rate (BRHR) introduced by Li et al. (2026a). BRHR measures the agreement between residue importance scores and the structural proximity of residues between two interacting chains, such as the peptide and CDR3b. Given a percentile threshold $t \in (0, 1]$, we select the top t fraction of residues with the highest importance scores in $\mathbf{S}$. A selected residue is considered a *hit* if its interaction distance also falls within the top t fraction of structurally interacting residues. We calculate the hit rate separately for each input sequence type in each sample and average the results across the TCR-XAI2 benchmark to obtain the final BRHR. A higher BRHR indicates better agreement between the model interpretation and the structurally defined binding regions. Following Li et al. (2026a), we report BRHR@.25, which measures the hit rate among the top 25% of residues with the highest importance scores and the 25% nearest residues based on structural proximity.

A.5 TCR-Epitope Prediction Models

To investigate how different input modalities and model designs affect TCR-epitope model learning, we analyze four models using EFT including NetTCR-2.2 (Jensen & Nielsen, 2023), MixTCRpred (Croce et al., 2024), TULIP (Meynard-Piganeau et al., 2024), and TCR-SRIM (Li et al., 2026b). NetTCR-2.2 is a CNN-based black-box model, for which we use Grad-CAM (Selvaraju et al., 2017) for post-hoc interpretation. MixTCRpred is an encoder-only Transformer model, and we use AttnLRP (Achtibat et al., 2024) to interpret its attention-based evidence. Since TULIP is an encoder-decoder Transformer, we use QCAI (Li et al., 2026a) to interpret its cross-attention. TCR-SRIM is an interpret-by-design model that directly provides contact maps between the peptide and CDR3a and between the peptide and CDR3b.

To examine whether the choice of post-hoc interpretation method affects the resulting explanations, we additionally apply AttnLRP to TULIP and Grad-CAM to MixTCRpred. A detailed analysis of how different post-hoc methods affect model interpretations is provided in the appendix. For the main analysis, we use the interpretation method that best matches the architecture of each model. For MixTCRpred, we use the model variant that includes the peptide as part of the input. TULIP provides two variants, with MHC allele information (TULIP-MHC) and without MHC allele information (TULIP-NoMHC). We analyze both variants to investigate how MHC allele information affects model explanations. TCR-SRIM is evaluated without structure regularization to provide a

consistent comparison with the other sequence-based models, using ESM2-8M as the backbone for computational efficiency. We additionally analyze structure-regularized TCR-SRIM (ESM2-8M) using different predicted structures to investigate how structural information affects model learning.

Table 3: **The final performance comparison of all models on IMMREP23, IMMREP25, and their respective original test datasets.** ROC-AUC at a false positive rate below 0.1 (ROC-AUC@FPR<0.1) indicates that all models converged to performance comparable to that reported in their original publications during our retraining.

Model	ROC-AUC@FPR<0.1			ROC-AUC
	IMMREP23	IMMREP25	Original Test	Original Test
MixTCRpred	0.611	0.513	0.746	0.850
NetTCR-2.2	0.575	0.516	0.748	0.886
TULIP-MHC	0.574	0.745	0.607	0.710
TULIP-NoMHC	0.587	0.744	0.632	0.736
TCR-SRIM	0.580	0.592	0.624	0.641

For training, we train each model three times using different random seeds (0, 1, and 2). Each model is trained on its original training dataset following the default training configuration specified by its original implementation. MixTCRpred is trained for 50 epochs, with interpretation tracking performed every epoch. NetTCR-2.2 is trained for 200 epochs, with interpretations tracked every 4 epochs. TULIP is trained for 300 epochs, with interpretations tracked every 6 epochs for both the MHC and NoMHC variants. TCR-SRIM is trained for 200 epochs, with interpretations tracked every 4 epochs for both the structure-regularized and non-regularized variants.

We primarily discuss the training trajectory with seed 0 in the main text and analyze variations across random seeds in the appendix. The final performance on IMMREP23 (Nielsen et al., 2024), IMMREP25 (IMMREP25, 2025), and the test datasets provided by the respective models is reported in Table 3. We report both ROC-AUC and ROC-AUC at a false positive rate below 0.1 (ROC-AUC@FPR<0.1), following the evaluation protocol used by IMMREP. We do not include the Assessment benchmark (Lu et al., 2026) because the evaluated models require different input modalities, resulting in two distinct groups of models that are not directly comparable. The results show that all models converge to performance levels consistent with those reported in their respective original papers.

A.6 Training Random Seed Variants

As model learning trajectories may vary across random seeds, we compare the explanations obtained from three random seeds for TULIP-NoMHC with QCAI and NetTCR-2.2 with GradCAM. As shown in Fig. 9, NetTCR-2.2 exhibits consistent explanations across seeds, primarily focusing on peptide-MHC interactions and, among the CDR regions, mainly CDR3a, CDR3b, and CDR1b. TULIP-NoMHC also shows similar explanations across seeds. Notably, all seeds exhibit a cluster corresponding to the CDR3b geometric region during the early stages of training. These results suggest that the models learn similar interpretation trajectories despite differences in random initialization.

A.7 Interpretation Method Difference

We further examine how different interpretation methods affect EFT explanations for the same model. As shown in Fig. 10 and Fig. 11, we compare GradCAM and AttnLRP for MixTCRpred, and QCAI and AttnLRP for TULIP-MHC. The results show that different interpretation methods can produce different explanations for the same model, potentially due to differences in interpretation quality. Therefore, we primarily analyze each model using its corresponding interpretation method that provides the most reliable explanations.

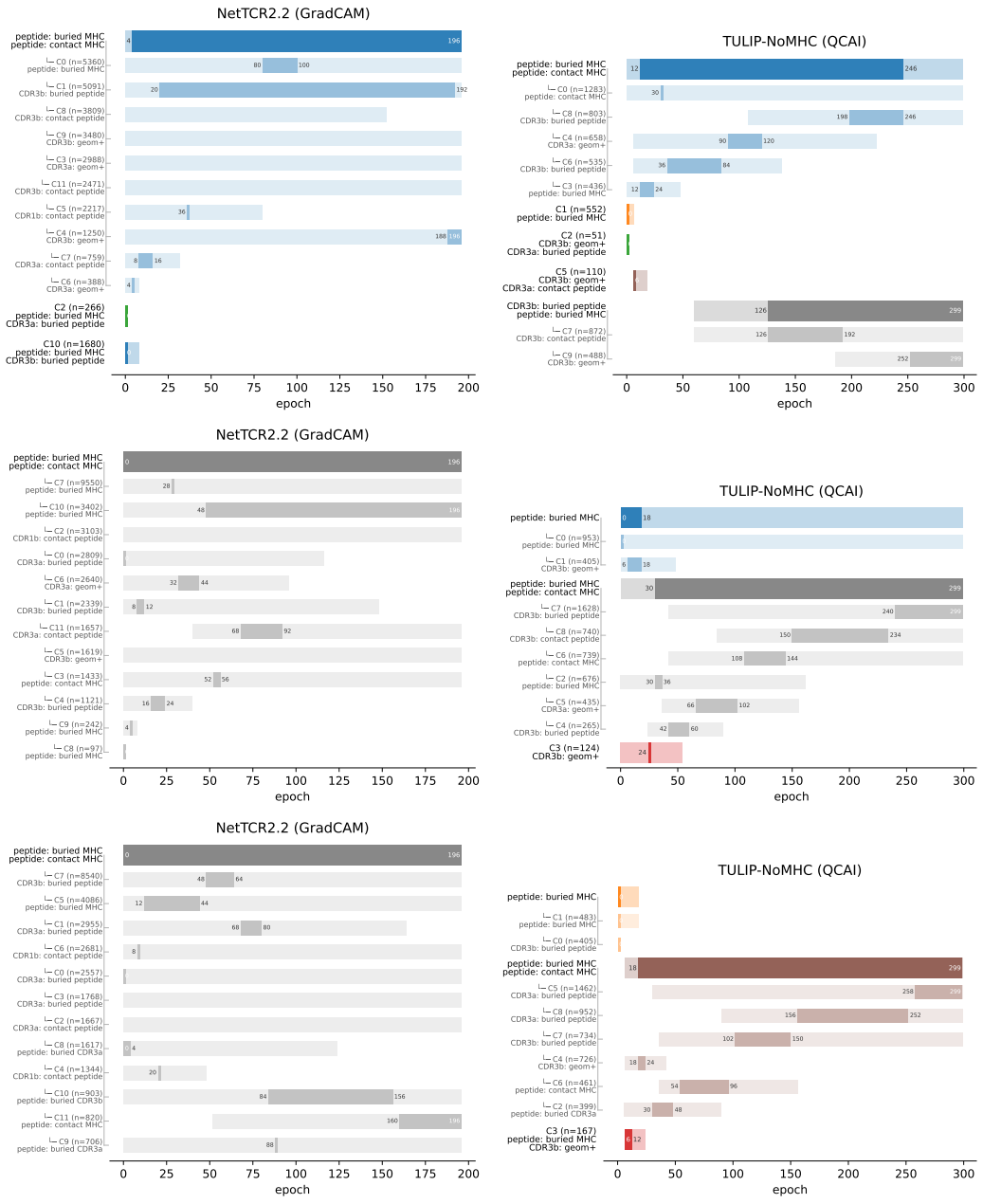

Figure 9: Different random seeds run for TULIP-NoMHC with QCAI and NetTCR-2.2 with Grad-CAM.

A.8 Different Allele-Specific Subsets

As we observe that training TCR-epitope models on allele-specific subsets may reduce conflicts between TCR α- and β-chain information and enable the model to learn more diverse information, we further compare models trained on different allele-specific subsets to examine this observation. As shown in Fig. 12, the learning trajectories across all three alleles exhibit similar patterns. These results suggest that allele-specific training may enable models to learn from a broader range of input features and coordinate information from the TCR α and β chains, thereby reducing conflicts between them.

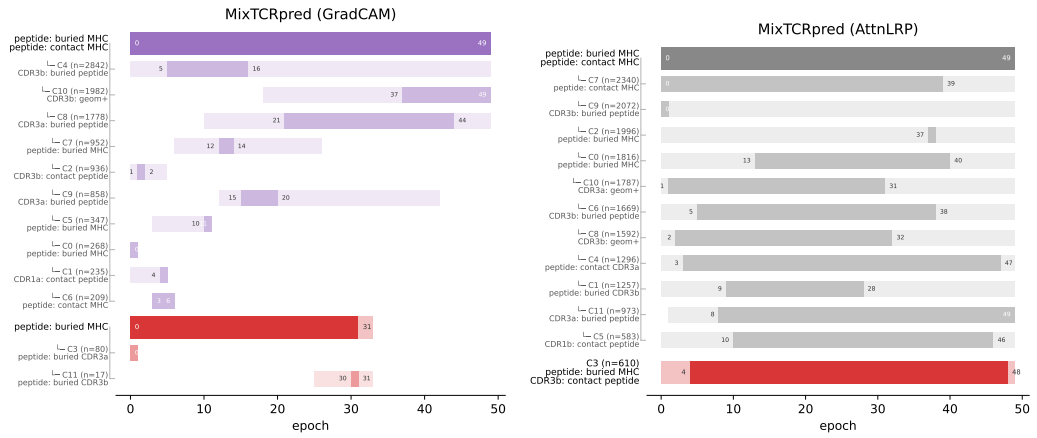

Figure 10: EFT explanation for MixTCRpred interpreted by GradCAM and AttnLRP respectively.

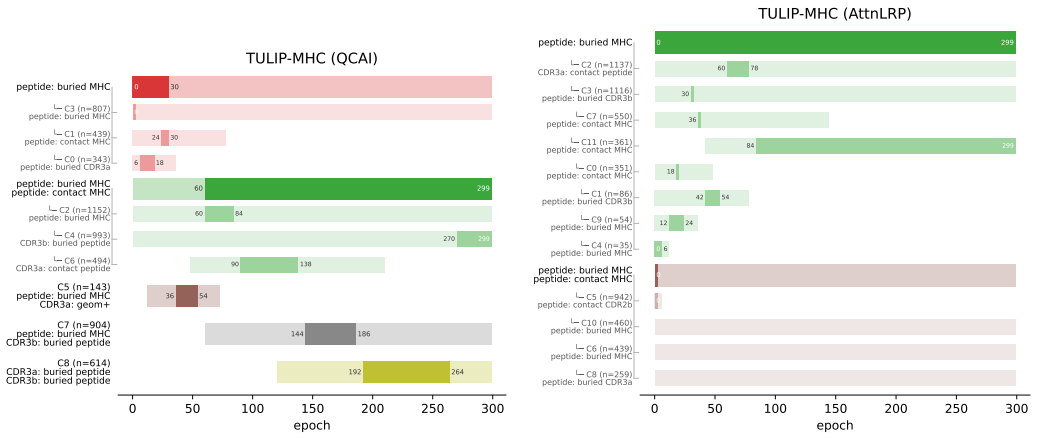

Figure 11: EFT explanation for MHC-TULIP interpreted by QCAI and AttnLRP respectively.

A.9 Explain Predicted Structure Difference in Guiding Model Training

Recent studies have explored incorporating structural information into TCR-epitope prediction models (Deleuran & Nielsen, 2025; Li et al., 2026b). Motivated by these approaches, we use the TCR-XAI2 benchmark with structures predicted by different structure prediction models to investigate how structural quality affects model learning. As a baseline, we first examine the evolution of BRHR in TCR-SRIM without structural regularization. As shown in Fig. 13, the BRHR trajectories of all chains fluctuate substantially throughout training rather than converging to stable patterns, indicating that the model's structural evidence remains unstable when no structural guidance is provided.

We then regularize TCR-SRIM using either experimentally resolved or predicted structures from the TCR-XAI2 training split and evaluate the evolution of BRHR on the test split using experimentally resolved structures. As shown in Fig. 14 and Fig. 15, when experimentally resolved structures are used for regularization, the BRHR of all three chains converges to stable values within approximately 70 epochs, with substantially reduced fluctuations. Predicted structures can also stabilize model interpretability, but their effects vary with structural quality. TCRModel2, which is adapted from AlphaFold2, exhibits a pattern similar to AlphaFold3. However, it enables TCR-SRIM to achieve higher CDR3a-peptide interpretability, reaching a BRHR of approximately 0.7, whereas AlphaFold3 reaches approximately 0.5, similar to the model regularized with experimentally resolved structures. Both AlphaFold-based structures nevertheless stabilize the model's interpretation of the three chains relatively quickly, consistent with their relatively high structural prediction

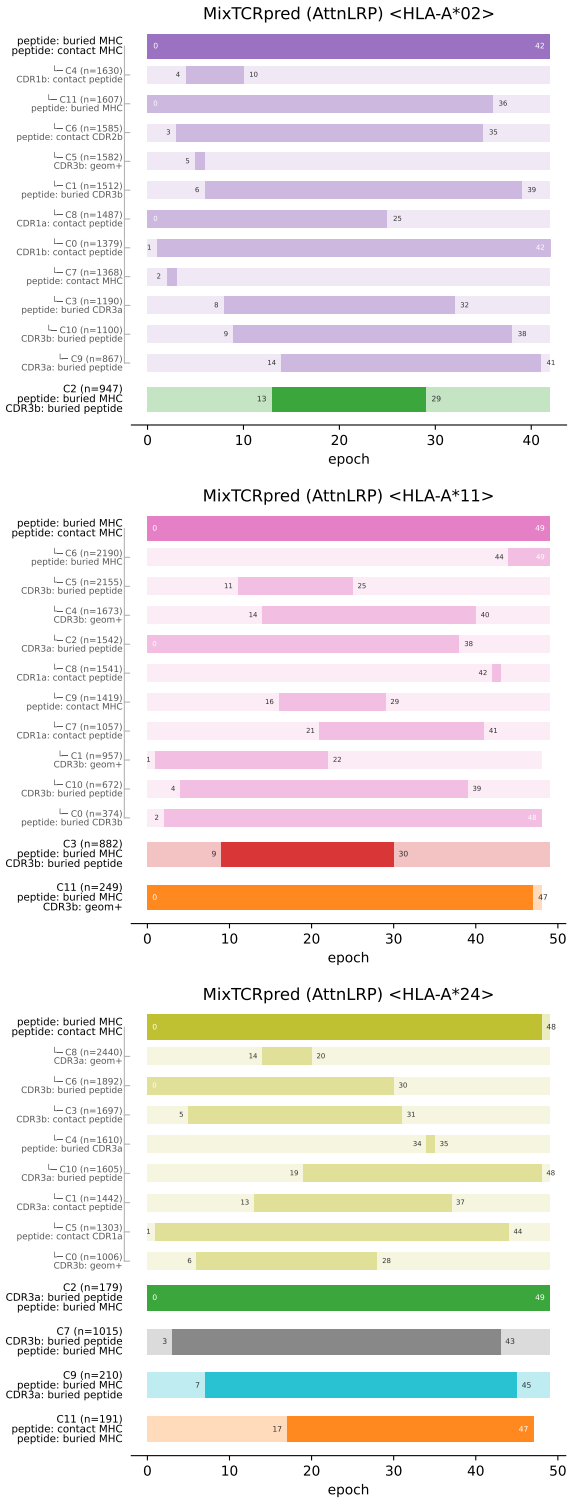

Figure 12: The EFT explanation for MixTCRpred interpreted by AttnLRP on `HLA-A*02`, `HLA-A*11`, and `HLA-A*24` specific subset.

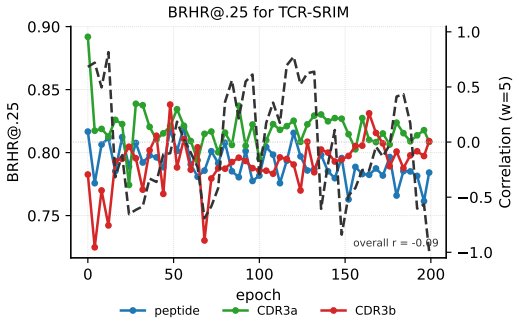

Figure 13: **The evolution of BRHR@0.25 during training for TCR-SRIM without structural regularization.** The model's interpretation remains highly unstable throughout training, with substantial fluctuations in BRHR across the three interaction regions.

quality. Although Boltz-2 also achieves high structural prediction quality, it produces a different training trajectory. Compared with the AlphaFold-based models, Boltz-2 leads to higher BRHR for CDR3b-peptide interactions than for CDR3a-peptide interactions, while the latter remains less stable throughout training. For tFold-TCR, which is designed for high-throughput structure prediction, the lower structural quality compared with the AlphaFold-based models and Boltz-2 is accompanied by less stable BRHR trajectories. OpenFold3 produces substantially lower-quality structures than tFold-TCR, and the corresponding model interpretation is also less stable. Overall, these results suggest that structural information can stabilize the model's understanding of TCR-epitope interactions during training, while the quality of the structural guidance influences both the stability and the interaction-specific patterns learned by the model.

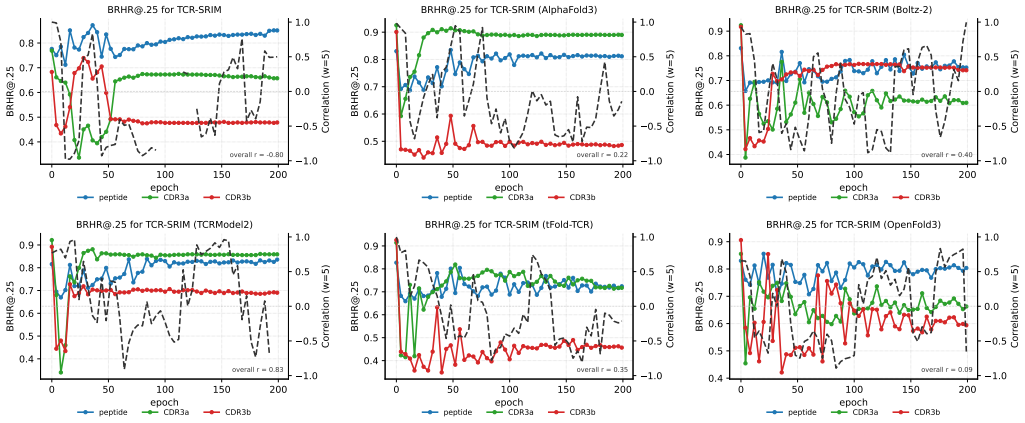

Figure 14: **The evolution of BRHR@0.25 during training for TCR-SRIM regularized with experimentally resolved structures and structures predicted by AlphaFold3, Boltz-2, TCR-Model2, tFold-TCR, and OpenFold3.** Structural regularization stabilizes the model's interpretation of CDR3-peptide interactions, with the degree of stabilization varying according to the quality of the structural guidance.

A.10 Insights for TCR-Epitope Model Design

According to our analysis of EFT across different TCR-epitope prediction models, and motivated by the finding that interpretability can provide guidance for model design (Li et al., 2025a), our results offer several insights for future TCR-epitope prediction models. For models without explicit MHC information, architectural mechanisms that facilitate coordination between the TCR alpha and beta chains, such as cross-attention between the two chains, may help reduce the conflicts observed during training. Another possible strategy is to first learn the pairing between the alpha and beta

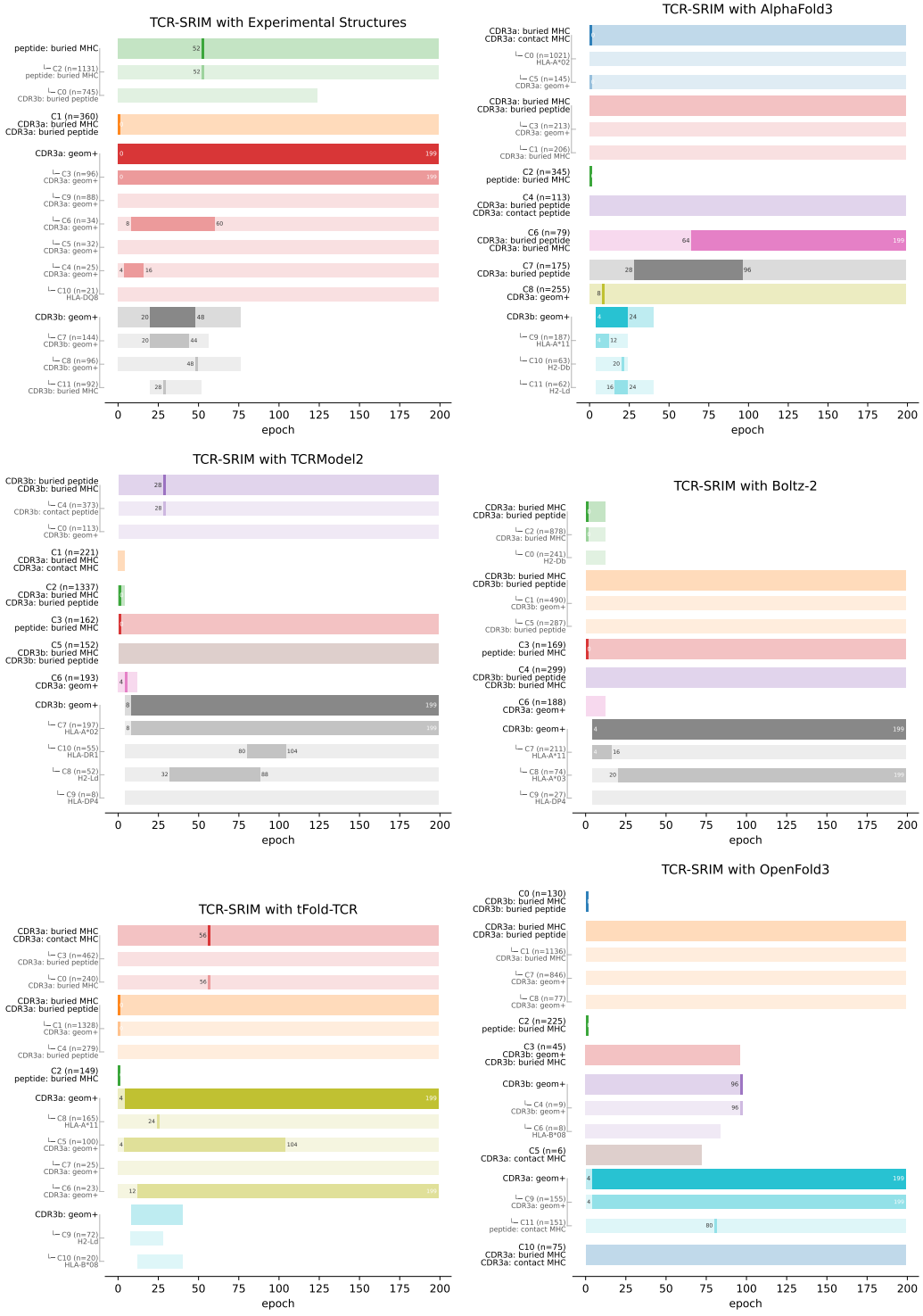

Figure 15: EFT for TCR-SRIM regularized with experimentally resolved structures and structures predicted by AlphaFold3, Boltz-2, TCRModel2, tFold-TCR, and OpenFold3.

chains independently before incorporating peptide information, which may encourage the model to establish coordinated TCR representations and reduce subsequent conflicts.

For structure-based models, our results suggest two important considerations. First, incorporating structural information can stabilize the model's interpretation during training. Second, the quality of predicted structures matters for how effectively they guide model learning. One possible direction is therefore to combine structures from different prediction models to provide complementary structural information, for example, using Boltz-2 and AlphaFold-based structures to better capture CDR3b-peptide and CDR3a-peptide interactions, respectively.