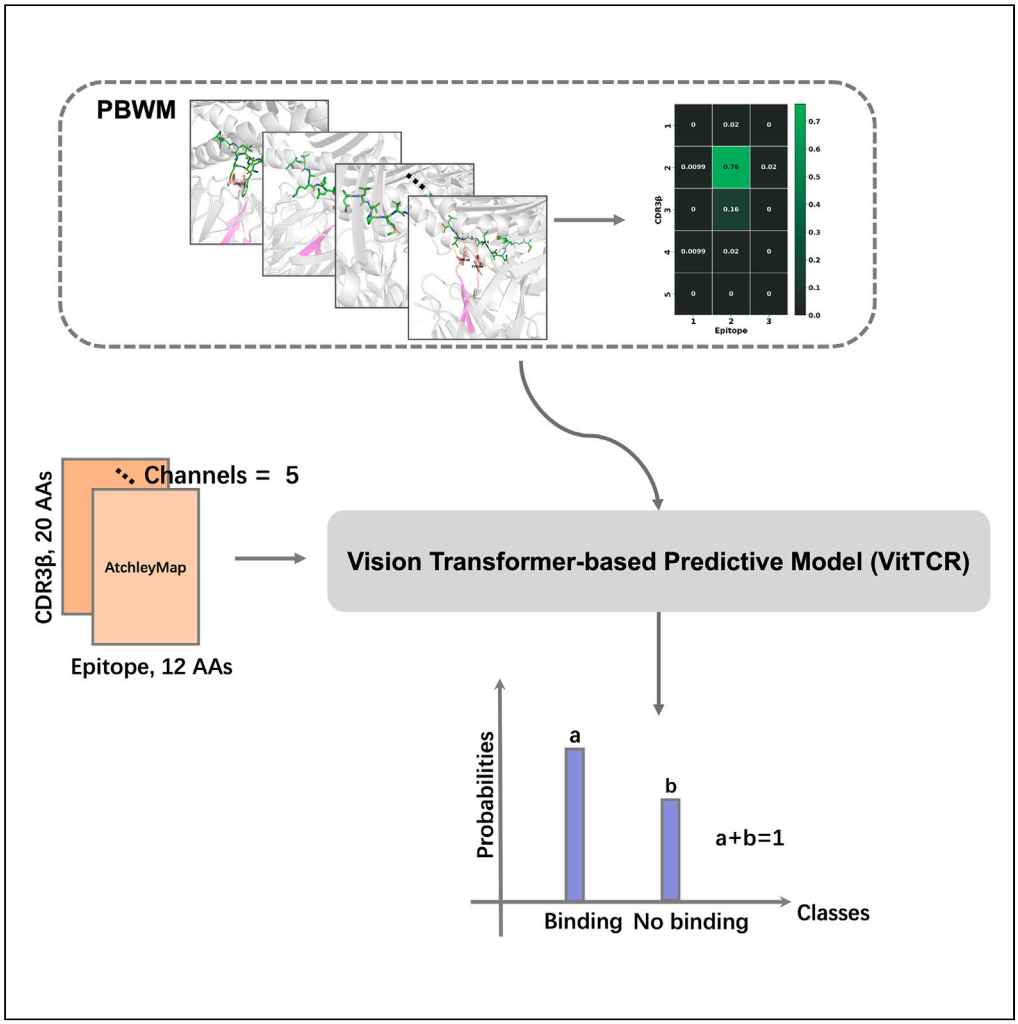
iScience

CellPress
OPEN ACCESS

Article

VitTCR: A deep learning method for peptide recognition prediction



Mengnan Jiang, Zilan Yu, Xun Lan

xlan@tsinghua.edu.cn

Highlights

VitTCR employs the structure of ViT for TCR-peptide interaction predictions

VitTCR offers the option to add PBWM, enhancing accuracy through interaction hotspots

TCR-epitope interactions mainly occur between AAs 5–16 of CDR3βs and 5–8 of epitopes

Jiang et al., iScience 27, 109770
May 17, 2024
https://doi.org/10.1016/j.isci.2024.109770

iScience

CellPress
OPEN ACCESS

Article

VitTCR: A deep learning method for peptide recognition prediction

Mengnan Jiang,[1,5] Zilan Yu,[1,2,5] and Xun Lan[1,2,3,4,6,*]

SUMMARY

This study introduces VitTCR, a predictive model based on the vision transformer (ViT) architecture, aimed at identifying interactions between T cell receptors (TCRs) and peptides, crucial for developing cancer immunotherapies and vaccines. VitTCR converts TCR-peptide interactions into numerical AtchleyMaps using Atchley factors for prediction, achieving AUROC (0.6485) and AUPR (0.6295) values. Benchmark analysis indicates VitTCR's performance is comparable to other models, with further comparative studies suggested to understand its effectiveness in varied contexts. Additionally, integrating a positional bias weight matrix (PBWM), derived from amino acid contact probabilities in structurally resolved pMHC-TCR complexes, slightly improves VitTCR's accuracy. The model's predictions show weak yet statistically significant correlations with immunological factors like T cell clonal expansion and activation percentages, underscoring the biological relevance of VitTCR's predictive capabilities. VitTCR emerges as a valuable computational tool for predicting TCR-peptide interactions, offering insights for immunotherapy and vaccine development.

INTRODUCTION

T cell receptors (TCRs) are heterodimers immobilized on the surface of T cells that recognize antigenic peptides presented by the major histocompatibility complex (MHC). Ninety-five percent of T cells are composed of highly variable α and β subunits linked by disulfide bonds, which are named αβ T cells. The remaining T cells are called γδ T cells and consist of γ and δ subunits.[1] The diversity of TCRs is mainly derived from the V(D)J recombination of immunoglobulin genes, with α and γ subunits arising from VJ recombination and β and δ subunits forming from VDJ recombination. Due to somatic recombination and random insertion of nucleotides during T cell development, humans have a highly diverse TCR repertoire, containing 10^{15} [2] to 10^{61} [3] possible receptors. The variable (V) region of the TCR subunits is responsible for the recognition of peptide-MHCs (pMHCs). The V region contains three highly variable complementarity-determining regions (CDRs): CDR1, CDR2, and CDR3. CDR3 is responsible for direct contact with antigenic peptides, which play an essential role in the recognition process. CD4 and CD8 are proteins expressed on the membrane surfaces of helper T cells and cytotoxic T cells, respectively. They can enhance the sensitivity and responses of T cells to pMHC.[4,5] This study focused on the CDR3 region of the TCRs of cytotoxic $CD8^+$ T cells.

Previous studies have mainly focused on the binding between MHCs and antigenic peptides, with methods such as ACME,[6] NetMHCpan-4.0,[7] DeepLigand,[8] MHCflurry,[9] DeepSeqPan,[10] MHCSeqNet,[11] and DCNN,[12] all of which are trained on the affinity data between MHCs and antigenic peptides. Currently, there is increasing attention being focused on the issue of TCR and epitope recognition. However, effective learning of TCR-pMHC recognition remains challenging due to the lack of a training dataset. Fortunately, with the development of sequencing technology, many more experimental datasets have been generated, including VDJdb,[13] IEDB,[14] and McPAS-TCR.[15] Several studies have shown that TCRs with similar CDR3 sequences are more likely to recognize the same peptide. While TCR classification methods, such as TCRdist,[16] GLIPH,[17] and TCRGP,[18] mainly focus on TCR sequences, interaction prediction methods, such as NetTCR2.0,[19] TITAN,[20] ImRex,[21] ERGO,[22] pMTnet,[23] and PanPep,[24] take peptide sequences into account. These models encode the AA sequences of TCRs and antigenic peptides separately and then concatenate them for feature extraction to predict whether an interaction occurs in a TCR-peptide pair.

Building a precise and robust model to predict TCR-peptide interactions remains challenging due to the high diversity of the TCR repertoire and various technological limitations. For instance, an antigenic epitope can be recognized by multiple T cell clonotypes,[16,17,25] while a T cell clonotype can exhibit cross-reactivity to multiple antigenic epitopes.[26] In addition, both α and β chains are considered to contribute to the binding specificity of TCR peptides.[18,19] However, despite the advent of single-cell TCR paired-strand sequencing, currently available TCR epitope binding data still mainly consist of single-strand (β-strand) information.

[1]School of Medicine, Tsinghua University, Beijing 100084, China
[2]Centre for Life Sciences, Tsinghua University, Beijing 100084, China
[3]Tsinghua-Peking Center for Life Sciences, MOE Key Laboratory of Tsinghua University, Beijing, China
[4]MOE Key Laboratory of Bioinformatics, Tsinghua University, Beijing 100084, China
[5]These authors contributed equally
[6]Lead contact
*Correspondence: xlan@tsinghua.edu.cn
https://doi.org/10.1016/j.isci.2024.109770

In this study, we leveraged the architecture of ViT,[27] initially designed for image recognition, to develop a novel model named VitTCR. VitTCR encodes the CDR3β-peptide interactions into AtchleyMaps and then takes the AtchleyMaps as inputs and outputs the predicted binding probabilities. Our results suggest that VitTCR achieved comparable performance with existing models when predicting interactions between HLA-A*02:01-restricted epitopes and CDR3βs.

RESULTS

The architecture of VitTCR

We have developed VitTCR, a deep learning-based method specifically designed to predict the interactions between epitopes and the CDR3β region of TCRs. Before making predictions for a specific pair of CDR3β and antigenic epitopes, it is essential to convert their alphabetical sequence information into numerical representations. In this research, the Atchley factors[28] were utilized as a method of numerical embedding for this purpose. The Atchley factors comprise five factors that encompass diverse physicochemical characteristics, enabling each amino acid (AA) to be effectively represented by these five factors. For each CDR3β-epitope pair, VitTCR translates it into a 3-dimensional numerical tensor referred to as AtchleyMap (STAR Methods). The encoding method we adopted is inspired by Intermap, as proposed by ImRex.[21] Both methods encode the sequences of CDR3β and antigenic epitopes into a three-dimensional vector. The difference, however, lies in the selection of amino acid indexes: while ImRex manually selects these based on factors that might affect the interaction, VitTCR adopts a broader range of Atchley Factors. As illustrated in Figure 1A, AtchleyMap has a width of 12 AAs, a height of 20 AAs, and a channel of 5, capturing the relevant information from the CDR3β and epitope sequences. The selection of the specific lengths (20 AAs for CDR3β sequences and 12 AAs for epitopes) is based on the results of statistical analysis (Figure S1), which shows that approximately 98.89% of CDR3β sequences fall within the length range of 10–20 AAs, and 97.97% of MHC-I-presented epitopes fall within the length range of 8–12 AAs. Therefore, setting the lengths to 20 AAs and 12 AAs ensures that the majority of sequences are properly represented. To provide a more detailed description of the interactions, we partitioned the AtchleyMaps into distinct patches and assigned numerical labels to each patch accordingly (Figure 1B, STAR Methods). As depicted in Figure 1C, VitTCR takes AtchleyMaps as inputs and generates predicted probabilities for "binding" or "no binding", with the two predicted probabilities summing to 1. Notably, the patches numbered in Figure 1B correspond to the patches numbered in Figure 1C to ensure consistent interpretation.

Performance comparison with other methods

Following model optimization, we then conducted a comparative analysis between VitTCR and other published methods. To ensure a fair comparison process, we employed the same training and test sets (Figure S2A), performed a 5-fold cross-validation with five iterations for each method, and evaluated the classification performance of the trained models on the same independent test set. Thus, each method produced 25 predicted values for the same test set, enabling us to statistically determine whether there are significant differences in the performance of different methods. As shown in Figure 2A, VitTCR performs better than these methods, including NetTCR-2.0 and ERGO_AE, in terms of the area under the receiver operating characteristic (AUROC) (median value: VitTCR = 0.6295, NetTCR-2.0 = 0.6040, ERGO_AE = 0.5816) and the area under the precision-recall curve (AUPR) (median value: VitTCR = 0.6485, NetTCR-2.0 = 0.5928, ERGO_AE = 0.5735). This suggests that our predictive model shows potential in predicting interactions between epitopes and the CDR3β region of TCRs.

After an exhaustive examination of the datasets, we found that the dataset used as an independent test set, VDJdb, contained a large number of unseen epitopes, while the proportion of seen epitopes was relatively low (see Figure S3), despite our adoption of a pair-based strategy for splitting the dataset. To enhance the accuracy of our results, we performed individual epitope-based comparisons and examined the model's performance on both seen epitopes (see Figures 2B, 2D, and 2E) and unseen epitopes (see Figures 2C, 2F, and 2G). In Figures 2B–2G, each data point corresponds to a subset of TCR-pMHC pairs in the test data involving a specific epitope. For one epitope, we generated 25 different AUPRs and AUROCs for each method with 5 repeated experiments of 5-fold cross validations. The mean values of these metrics are shown as data points in these plots. Both training and testing data are the same for all methods tested. As shown in these figures, all methods demonstrate better predictive performance with seen epitopes (Figure 2B) than with unseen epitopes (Figure 2C). In the case of seen epitopes, although VitTCR shows the highest median (Figure 2B) and more dominant antigenic epitopes (epitopes located in the upper left of the diagonal line) compared to the other two methods (Figures 2D and 2E), statistical analysis (t-test) indicates that this difference is not significant. However, in the dataset of unseen epitopes, VitTCR showed improved performance compared to the other two methods (Figure 2C) and identified a higher number of epitopes (Figures 2F and 2G).

Furthermore, the field focusing on the mutual recognition of TCRs and epitopes has evolved, with an increasing emphasis on evaluating models' predictive capabilities for unseen epitopes. To reflect this shift, we have updated Figure 2H to include the performance evaluation of additional models, such as DLpTCR, ImRex, TITAN and the more recent pMTnet and PanPep, specifically on datasets comprising unseen peptides. The independent dataset utilized here originated from the independent test set (from which seen epitope-related samples have been removed) of Figure 2, wherein samples that overlapped with the training sets of other models were removed. These non-overlapping samples were then used as a unified test set containing only unseen epitopes for this benchmark. Figure 2H shows that VitTCR achieved comparable AUPR and AUROC metrics to the evaluated models.

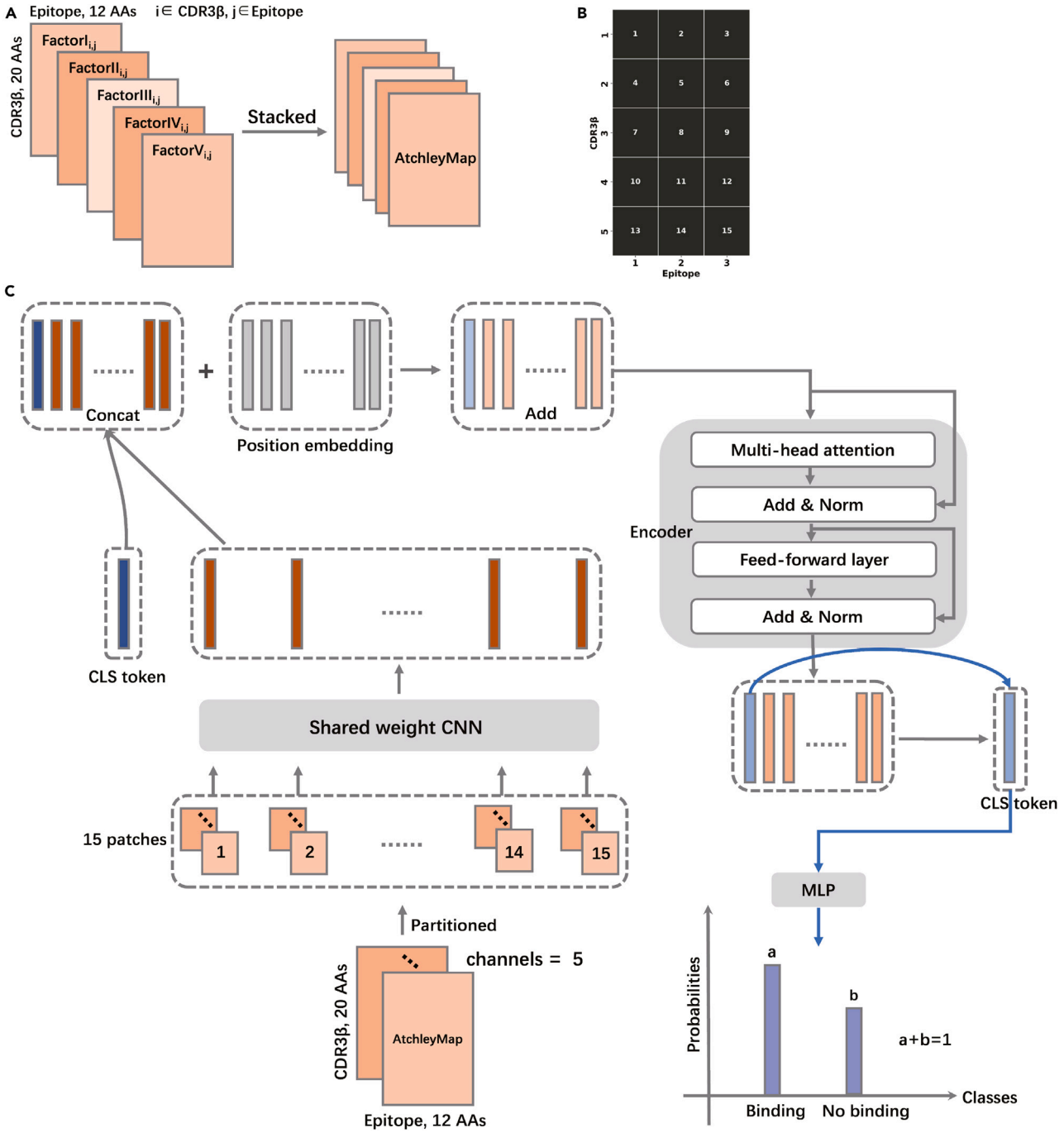


Figure 1. The architecture of VitTCR for predicting interactions between CDR3βs and epitopes

(A) Schematic diagram of AtchleyMap encoding. The value of each position was the absolute value of the difference in the Atchley factor of two corresponding AAs. The subscript i (ranging from 1 to 20) represents the position of amino acids of CDR3βs, and j (ranging from 1 to 12) represents the position of amino acids of epitopes.

(B) Strategy of patch division. Each AtchleyMap is partitioned into 15 patches.

(C) The architecture of VitTCR. VitTCR takes AtchleyMaps as inputs and generates predicted probabilities for "binding" or "no binding", with the two predicted probabilities summing to 1. AA: amino acid; CNN: convolutional neural networks; CLS: classification; MLP: multilayer perceptron.

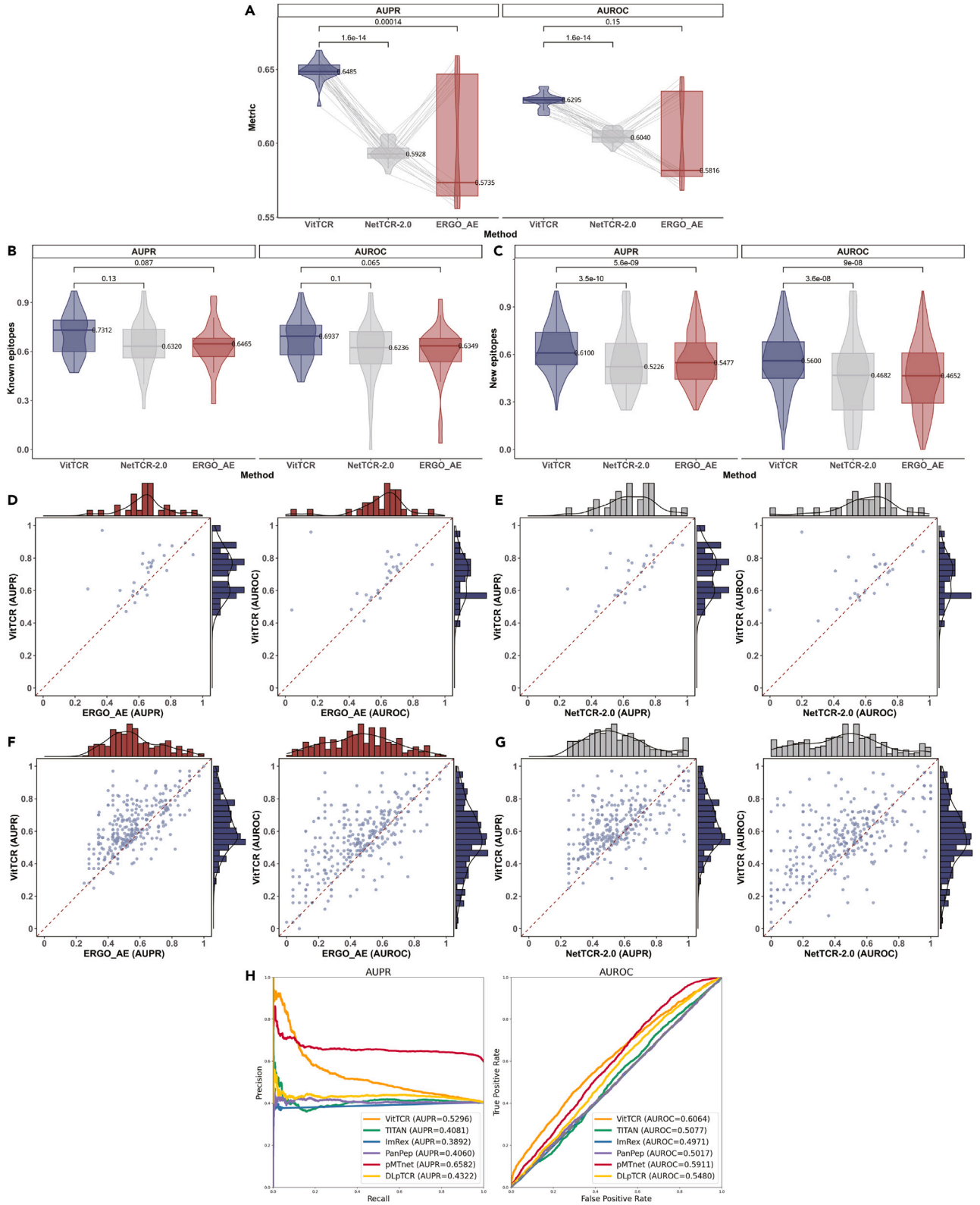

A
AUPR
AUROC
0.00014
0.15
1.6e-14
1.6e-14
Metric
0.6485
0.5928
0.5735
0.6295
0.6040
0.5816
VitTCR
NetTCR-2.0
ERGO_AE
Method
B
AUPR
AUROC
0.087
0.065
0.13
0.1
Known epitopes
0.7312
0.6320
0.6465
0.6937
0.6236
0.6349
C
AUPR
AUROC
5.6e-09
9e-08
3.5e-10
3.6e-08
New epitopes
0.6100
0.5226
0.5477
0.5600
0.4682
0.4652
D
E
F
G
VitTCR (AUPR)
VitTCR (AUROC)
ERGO_AE (AUPR)
ERGO_AE (AUROC)
NetTCR-2.0 (AUPR)
NetTCR-2.0 (AUROC)
H
AUPR
AUROC
Precision
Recall
True Positive Rate
False Positive Rate
VitTCR (AUPR=0.5296)
TITAN (AUPR=0.4081)
ImRex (AUPR=0.3892)
PanPep (AUPR=0.4060)
pMTnet (AUPR=0.6582)
DLpTCR (AUPR=0.4322)
VitTCR (AUROC=0.6064)
TITAN (AUROC=0.5077)
ImRex (AUROC=0.4971)
PanPep (AUROC=0.5017)
pMTnet (AUROC=0.5911)
DLpTCR (AUROC=0.5480)

Figure 2. Comparison of VitTCR with other published methods on an independent test set

(A) Comparison of model performance in terms of AUPR (left) and AUROC (right): All methods were trained and validated using identical datasets. For each method, five iterations of 5-fold cross-validation were conducted. Consequently, each model produced 25 predicted values for the same test set — five from each round of the 5-fold cross-validation. These values are represented in the resulting plots, with dark gray lines connecting the dots corresponding to each iteration. The values shown in the plots are medians.

(B) Performance of different methods on visible epitopes: In this analysis, each point represents an epitope and the values shown in the plots are medians.

(C) Performance of different methods on unseen epitopes, analogous to the analysis presented in Figure 2B.

(D) Model performance comparison between VitTCR and ERGO_AE on visible epitopes: Each spot represents an epitope, with the x axis representing the AUPR (left)/AUROC (right) for ERGO_AE and the y axis representing the AUPR (left)/AUROC (right) for VitTCR.

(E) Model performance comparison between VitTCR and NetTCR-2.0 on visible epitopes, analogous to the analysis presented in Figure 2D.

(F and G) Model performance comparison between VitTCR and other methods on unseen epitopes, analogous to the analysis presented in Figures 2D and 2E.

(H) Comparison of the predictive performance of different models on the same test set (left: AUPR, right: AUROC).

Paired t-tests were conducted and the p-values are annotated at the the top of each figure.

The influence of PBWM on model performance

To investigate positional bias and identify the crucial regions involved in the interactions between CDR3βs and epitopes, a set of 83 structurally resolved pMHC-TCR complexes was downloaded from the PDB database. The interacting AA pairs in these complexes were labeled using PyMOL (Figure 3A; Table S1). The next step involved counting the occurrences of interacting AA pairs within the downloaded complexes and normalizing the counts for each patch (Figure 1B). This normalization was achieved by dividing the counts by the total number of interacting AA pairs. This analysis provided a positional bias weight matrix (PBWM). Each value in the weight matrix represents the percentage of the total number of interacting AA pairs occurring in that particular patch, and a higher value in a specific patch indicated a greater likelihood of CDR3β-epitope interactions occurring in that particular region. As displayed in Figure 3B, the sum of the percentages across the 15 patches equals 1, and most interactions tended to occur between the middle patch of the epitopes and the second to fourth patches of the CDR3β regions. In other words, interactions are more likely to occur between the fifth to sixteenth AAs of CDR3βs and the fifth to eighth AAs of epitopes.

VitTCR provides a way of integrating PBWM (Figures 3C and S4, and STAR Methods). To determine whether the PBWM can improve the prediction, we added the weight matrix into VitTCR and then compared the results of VitTCR with or without the PBWM. As illustrated in Figure 3D, the performance of VitTCR showed improvement on the testing set after adding the calculated weights, suggesting that the weight matrix may capture the pattern of CDR3β-peptide interactions to a certain degree. To make the results more intuitive and detailed, we also performed comparative analyses based on individual antigenic epitopes (see Figures 3E and 3F). Figure 3E, where each dot represents an epitope, indicates that VitTCR with PBWM showed slight improvement in performance compared VitTCR without PBWM, in both AUPR (left) and AUROC (right) comparisons. Additionally, as depicted in Figure 3F, VitTCR with PBWM identified a higher number of epitopes (those located in the upper left of the diagonal). However, the improvement in model performance with the addition of PBWM is limited at the moment. With the availability of additional data, it is possible for further refinement and enhancement of the PBWM.

Cluster-based filtering can decrease false positives

For applications involving extensive experimental validation, it is essential to avoid false positive (FP) predictions since FP predictions will lead to unnecessary downstream validation and increased labor and time costs. Model sensitivity is usually not the main concern in practical applications for identifying TCR-peptide interactions. Instead, our primary goal is to increase the positive predictive value (PPV) of the model. The PPV is the proportion of true-positive (TP) results to the total number of predicted positive results (TP + FP). Thus, optimizing a model's PPV becomes critical. Dash et al.[16] found that CDR3βs with higher sequence similarities tend to recognize the same epitope. Therefore, we speculated that removing unclustered CDR3βs with distinct AA sequences from all other CDR3βs from the training dataset or testing dataset could improve the model performance. To test this hypothesis, we utilized iSMART[29] for CDR3β clustering and removed unclustered CDR3βs from the dataset (STAR Methods).

To investigate the effect of cluster-based filtering, we performed filtering operations on the training and test sets separately (Table S2). First, we classified the trained models into four major categories, including Original (neither the training set nor the test set was filtered), Trainset-only (only training set clustered and filtered), Testset-only (only test set clustered and filtered), and Clustered (both training set and test set clustered). Then, we conducted five repeated 5-fold cross validations under the four settings. These results suggest that performing cluster-based filtering either on the training set or on the test set significantly improved the PPV of the model on the independent testing set (Figure 4A, left panel). In addition to PPV, cluster-based filtering of datasets also significantly improved the AUROC (middle panel) and AUPR (right panel) for VitTCR. In addition, to ensure that this finding is not merely coincidental, we conducted the same analysis on other models, including NetTCR-2.0 and ERGO_AE. As shown in Figure S5, the application of cluster-based filtering to datasets appears to improve the PPV for our models as well as for NetTCR-2.0 and ERGO_AE, suggesting a potential benefit across different models. Additionally, the trends observed in AUROC and AUPR for NetTCR-2.0 and ERGO_AE seem to align with those observed for VitTCR.

Considering the possibility that cluster-based filtering may result in data related to epitopes with fewer cognate CDR3β being filtered out, thereby leading to improved model performance, we performed statistical analyses (STAR Methods) on the datasets before and after filtering to rule out this possibility. We first counted the CDR3β for each epitope in the dataset before filtering, then ranked these counts in descending order. The ranking was represented using percentiles, where lower percentile values indicate higher CDR3β counts (and higher epitope

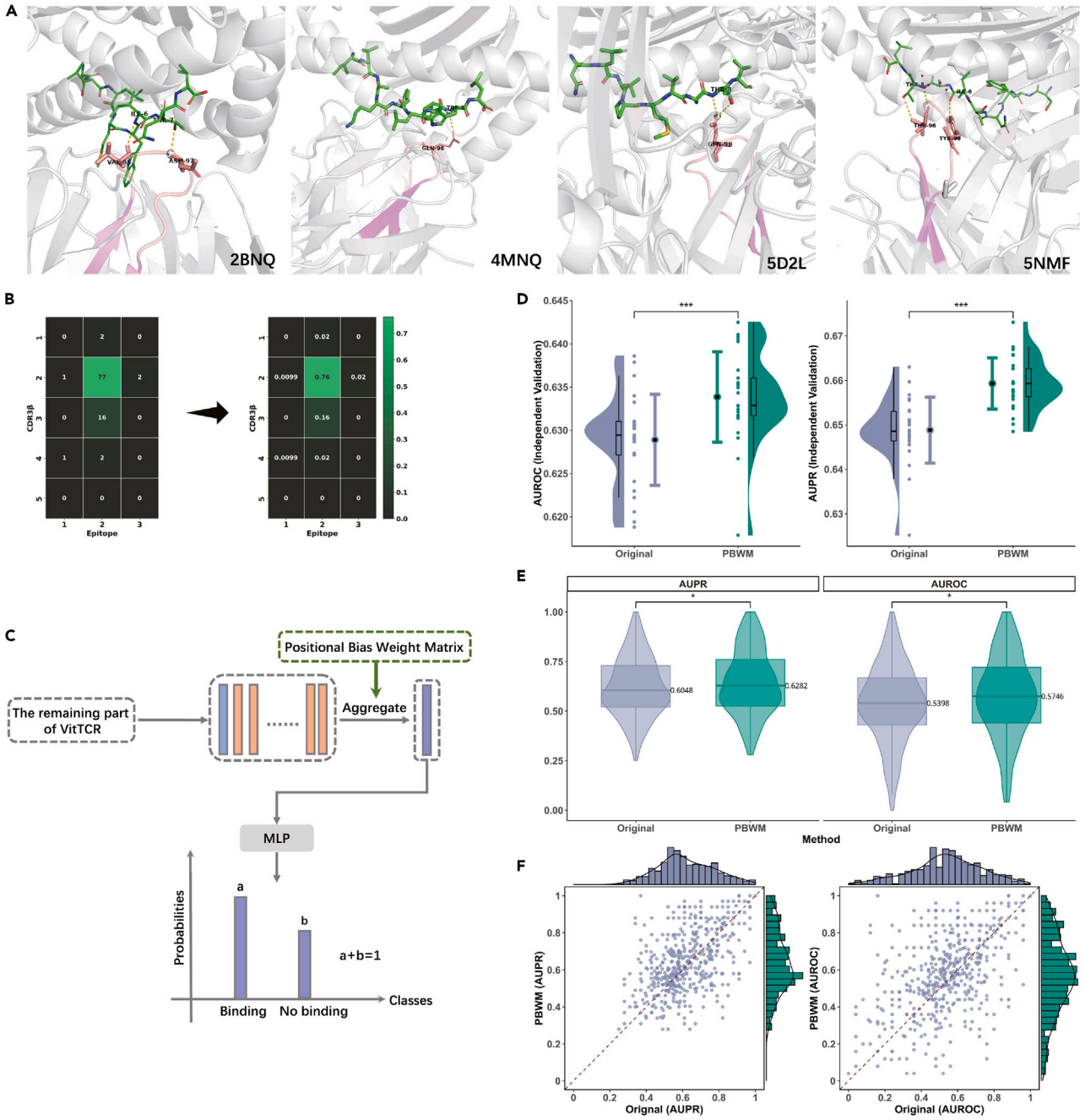


Figure 3. The effects of PBWM on the performance of VitTCR

(A) Four pMHC-TCR complexes (PDB: 2BNQ, 4MNQ, 5D2L, and 5NMF) are labeled and colored using PyMOL. The complexes are visualized with specific color schemes: yellow dotted lines represent polar interactions in CDR3β-peptide pairs, the CDR3 region of TCRβ chains is colored pink, and the epitopes are colored green.

(B) The left panel illustrates the count of interacting AA pairs, where one AA is from the CDR3β region of the TCR and the other is from the epitope. A total of 101 AA pairs with interactions were identified. In the right panel, the matrix from the left panel is normalized by dividing the count in each patch by the total number of interacting AA pairs.

(C) Integration of PBWM with VitTCR.

(D) Comparison of the performance of VitTCR in the independent test set before and after integration of the PBWM.

(E) Performance comparison of VitTCR for each epitope before and after the integration of the PBWM. Each point represents an epitope, and the values shown in the plots are medians.

(F) Performance comparison for each epitope, with the x axis representing the AUPR (left)/AUROC (right) of the original VitTCR, and the y axis representing the AUPR (left)/AUROC (right) of VitTCR integrated with PBWM. This is analogous to the analysis presented in Figures 2D–2G.

All p-values were determined using paired t-tests.

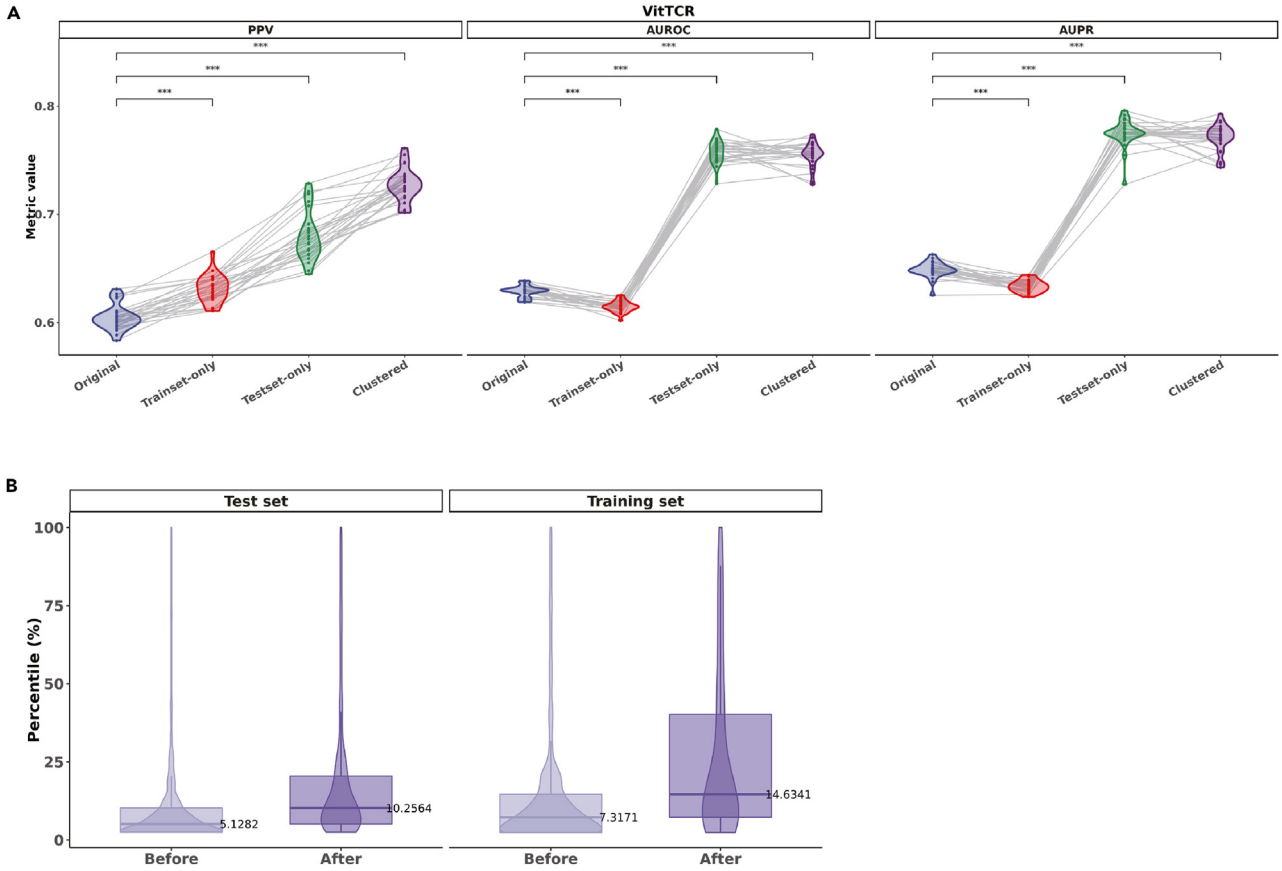


Figure 4. The influence of cluster-based filtering on model performance

According to the dataset configuration, the model was divided into four main categories: Original, Trainset-only, Testset-only and Clustered. And the significance was determined using paired t-tests.

(A) The PPVs (left), AUROCs (middle), and AUPRs (right) of VitTCR under different dataset configurations were compared. Five iterations of 5-fold cross-validation were conducted for the three configurations, and each dot in this figure represents a fold replicate, with dark gray lines connecting the dots corresponding to each iteration.

(B) The distribution of percentile values for epitopes in the dataset before (Before, left panel) and after (After, right panel) filtering. Each point in the violin plot corresponds to an individual epitope, and the values depicted in the figure represent the medians.

rankings), and higher percentile values indicate lower CDR3β counts. After that, we extracted all epitopes from the post-filtering dataset, assigning them the percentile ranks based on the initial, unfiltered dataset. To visualize the impact of filtering, we plotted the distribution of these percentile ranks for epitopes before and after filtering (Figure 4B). As shown in Figure 4B, contrary to expectations of a downward trend, the median percentile value in the filtered dataset actually shows an upward trend, indicating no overall shift toward epitopes with more CDR3βs after filtering. In summary, our analysis suggests that the observed improvement in model performance does not appear to be solely attributable to the exclusion of epitopes associated with a smaller number of TCRs. It is important to note that the essence of filtering is to make the dataset more simple, thereby aiding in the identification of TCRs specific to epitopes.

Correlation between model predictions and TCR clone fraction

To verify model performance from a novel perspective, we conducted an investigation into the correlation between the model predictions and TCR clone fraction. Additionally, we conducted a comparative analysis of VitTCR with other methods, including ERGO_AE and NetTCR-2.0, using the same dataset. We obtained the single-cell TCR sequencing data of $CD8^+$ T cells of four healthy human donors from the 10x Genomics platform (10x Genomics). Given that the training dataset for our model was limited to HLA-A02:01 data points and HLA-A02:01 expression was observed only in Donor 1 and Donor 2, we focused the analysis on the data from these two donors. The dataset included a highly multiplexed panel consisting of 44 different pMHC multimers and 6 control pMHC multimers to determine the binding specificity of each $CD8^+$ T cell. The binding specificity between each T cell and each tested pMHC was quantified by counting the number of unique molecular identifier (UMI) sequences associated with that specific pMHC in the T cell. For each T cell, if the UMI count for any of the 44 pMHC multimers exceeded 10 and was more than five times higher than the highest UMI count for the 6 negative controls, the cell was

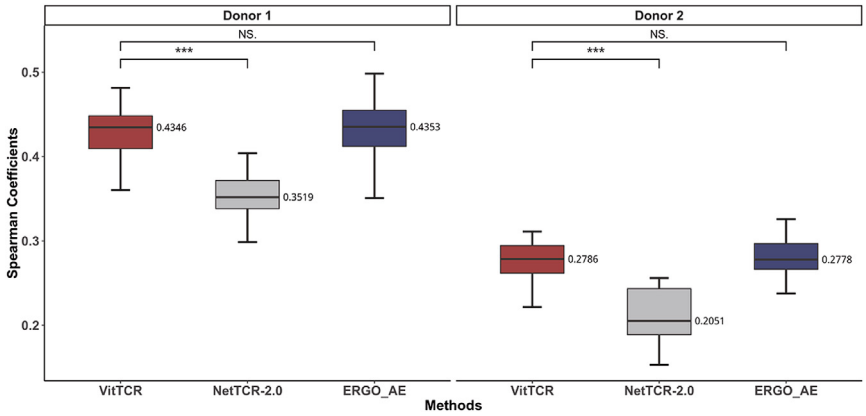


Figure 5. Correlation between predicted probabilities and clone fractions for different methods

Spearman correlation coefficients between predicted probabilities and clone fractions were calculated for VitTCR, NetTCR-2.0, and ERGO_AE using different healthy donor data. For all methods, five repetitions of 5-fold cross validation were performed. Each point of the boxplot represents the correlation coefficient between the predicted probabilities of the model and the clone fractions for each fold, and the number on the right side of the boxplot indicates the median correlation coefficient. The significance was determined using paired t tests.

considered to exhibit significant specificity toward that pMHC. During the cell filtration process, T cells showing significant specificity toward fewer than one pMHC or exceeding four pMHCs were excluded from the analysis. After the filtering of T cells, for each clonotype of TCR, we calculated the percentage of T cells harboring that specific clonotype based on the total number of T cells measured in the donor, irrespective of the HLA-A*02:01 restriction. This percentage represents the clone fraction associated with that particular TCR clonotype and provides insights into the clonal expansion of T cells. Subsequently, for each TCR clonotype, the pMHC with the highest UMI counts was chosen, and the binding probability between the CDR3β of the TCR clonotype and the epitope of the pMHC was predicted. Finally, we calculated the Spearman correlation coefficient between clone fractions and predicted binding probabilities.

To ensure a rigorous and unbiased comparison process, we performed five repetitions of 5-fold cross-validation for VitTCR, NetTCR-2.0 and ERGO_AE separately. Subsequently, we calculated the Spearman correlation coefficients between the predicted binding probabilities and the TCR clone fractions for each fold. This approach enabled us to objectively evaluate the associations between the predicted probabilities and the observed clone fraction for the methods mentioned above. As displayed in Figure 5, the predicted binding probabilities of all methods appear to have a weak positive correlation with the clone fraction, suggesting a potential relationship between these variables. While the correlation coefficients suggest a range from weak to medium, there appears to be a trend where higher predicted binding probabilities are associated with stronger clonal proliferation of T cells in the dataset. Each TCR clonotype was assigned 25 different predicted probabilities by VitTCR, NetTCR-2.0, and ERGO_AE. We calculated the mean predicted probability for each prediction method as the representative predicted probability for that particular TCR clonotype. Figure S6 displays the visualization of the mean predicted probabilities against the clone fractions. The relevant metrics are summarized in Table 1. Specifically, the correlation coefficients between VitTCR's predicted binding probabilities and clone fraction tended to be higher than those for NetTCR-2.0. The correlation coefficients for VitTCR and ERGO_AE were similar, showing no significant difference. Generally, the predictions of all methods demonstrated a correlation with clone fraction.

Correlation between model predictions and T cell activation

Furthermore, we sought to examine whether the predicted binding probabilities were positively correlated with the activation percentage of T cells. In a previous study (manuscript submitted), we cocultured 15 T cell clonotypes with 11 immunogenic peptides of SARS-CoV-19 in a pairwise manner and selected CD69 as a marker of T cell activation to quantify the percentage of T cell activation. Specific details on calculating the T cell activation percentage can be found in Methods. The analysis in this section is similar to that for clone fraction, with ERGO_AE and NetTCR-2.0 also being selected for analysis. In total, we obtained 165 CDR3β-epitope pairs. The aforementioned steps of coculture and quantification of T cell activation were repeated three times so that each CDR3β-epitope pair had three activation percentage values, and the mean value was taken as its percentage activation in this study.

As depicted in Figure 6A, the Spearman correlation coefficient between the predicted binding probabilities and the activation percentages of T cells was higher for VitTCR than for ERGO_AE, while there was no significant difference in performance between VitTCR and NetTCR-2.0. As shown in Figure 6B, CDR3β-epitope pairs with higher T cell activation percentages tended to have higher predicted binding probabilities by VitTCR. One thing to note is that only 4 bins are shown in Figure 6B. This may be due to the small sample size (comprising only 165 pairs), which resulted in the model's predictions not covering the entire range from 0 to 1. This observation tentatively suggests that the predicted results of VitTCR may partially reflect cell activation, hinting at a potential role in reflecting the activation status of T cells, though this interpretation is subject to further validation given the dataset's limitations. The predictions from all methods appear to correlate with the

Table 1. The Spearman correlation coefficients and *p* values between the mean of predicted probabilities and clone fractions for different methods

	Donor 1		Donor 2	
Model	correlation coefficient	*p* value	correlation coefficient	*p* value
VitTCR	0.47	1.1e-16	0.26	7.8e-07
NetTCR-2.0	0.40	1.9e-16	0.21	7.e7-05
ERGO_AE	0.46	1.3e-16	0.27	2e-07

activation percentage to some extent, which could lend support to the utility of these predictive approaches, pending further investigation. However, it's important to acknowledge that the Spearman correlation coefficients are weak. This indicate that the association between T cell activation and TCR-epitope recognition is not a simple linear process[31] and that the progress of T cell activation is highly context-specific. These findings point to the need for further analysis to better understand the implications of these correlations.

DISCUSSION

In this study, we developed a novel method named VitTCR to predict the interactions between epitopes and the CDR3β region of TCRs. The encoding method we adopted is inspired by Intermap, as proposed by ImRex.[21] For each CDR3β-epitope pair, VitTCR encodes it into a 3-dimensional numeric tensor named AtchleyMaps. The length of the tensor corresponds to the length of CDR3β (20 AAs), the width corresponds to the length of the epitope (12 AAs), and the channels represent the 5 Atchley factors. Subsequently, VitTCR takes the AtchleyMaps as inputs and outputs the predicted binding probability for each CDR3β-epitope pair.

VitTCR provides an option for adding PBWM. With the assistance of PBWM, VitTCR assigns greater weight to patches in regions more likely to interact. The performance of VitTCR showed slight improvement on the independent test set after adding the PBWM, hinting at the possibility that the physicochemical properties characterized by the Atchley Factors may influence the interaction process. Additionally, it suggests that the weight matrix may capture the pattern of CDR3β-peptide interactions to some extent. As more data appear, we anticipate that the positional bias weights could potentially improve the generalizability of the model. In this study, due to the insufficient paired data of TCR, our focus was primarily on CDR3β, despite recognizing the importance of CDR3α in peptide recognition. Including CDR3α in future analyses could potentially enhance the precision and reliability of predictions. Furthermore, we attempted to extend our model's applicability beyond the data range of HLA-A02:01, rather than restricting it to a single MHC type. However, initial results indicated that the model's performance other HLA types did not meet our expectations, highlighting the need for future efforts to gather more data for various HLA types

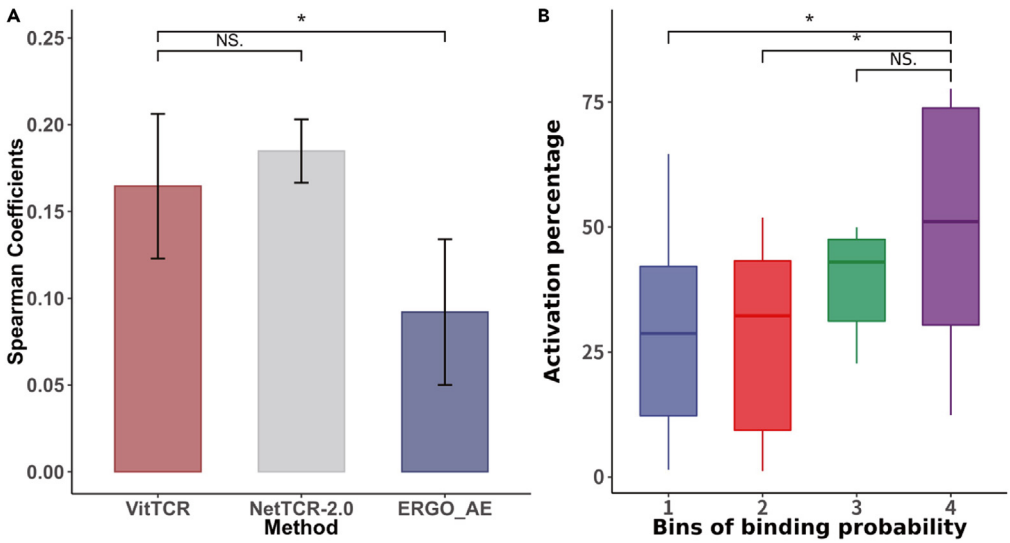


Figure 6. Correlation between predicted probabilities and activation percentages for different methods

(A) Spearman correlation coefficients between predicted probabilities and activation percentages of T cells were calculated for VitTCR, NetTCR-2.0, and ERGO_AE. For all methods, 5-fold cross-validation was performed. Each point of the boxplot represents the correlation coefficient between the predicted probabilities of the model and the activation percentages for each fold.

(B) The binding probabilities ranging from 0 to 1 were divided into five bins, with each bin representing an interval of 0.2. The x axis represents the bins of binding probabilities predicted by VitTCR, while the y axis represents the activation percentages of TCR clonotypes.

The significance was determined using paired t-tests.

and further refine the model for better performance across diverse HLA contexts. We remain hopeful that an increase in training data on TCR–pMHC interactions will not only make predictions more reliable but also provide an effective adjunct for vaccine design and tumor immunotherapy.

Limitations of the study

We acknowledge several significant limitations of the proposed model. First, the effectiveness of the VitTCR model is contingent upon the quality of the datasets utilized. High-quality datasets are essential for successful model development, but our current dataset presents considerable challenges for predictive methodology development. These challenges stem from both its limited size and significant imbalances. Second, although repeated 5-fold cross-validation has revealed significant differences (t-test, p-value <0.001), the observed improvements in model performance with the integration of PBWM are modest (as shown in Figure 3D). This suggests that expanding the dataset might be beneficial for refining PBWM and its role in improving prediction accuracy warrants further exploration with more diverse data. Third, the construction of negative data from TCRs not exclusively linked to HLA-A02:01 raises concerns about the model's potential bias toward MHC restriction recognition rather than epitope-peptide interactions.[32] Lastly, VitTCR model is currently optimized for HLA-A*02:01. Generalization of the model will likely involve enriching the training data with a wider array of relevant features, including CDR3α, VDJ gene usage, and various MHC-I subtypes, in addition to CDR3β.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Dataset processing
  - Statistical analysis of sequence length
  - Statistical analysis of HLA
  - Atchleymap encoding
  - Patch division
  - Architecture of VitTCR
  - Model training and selection
  - Self-attention mechanism of VitTCR
  - PBWM integration
  - Cluster-based filtering
  - Details of the validation of cluster-based filtering
  - T-cell activation percentage
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2024.109770.

ACKNOWLEDGMENTS

We thank Dr. Lihui Wang (Tsinghua University) for kindly providing the T cell activation-related experimental data. We also thank all members of the Lan lab for their helpful discussions and feedback. This work was funded by grants from the Tsinghua University Independent Research Project (ID: 52302102423), the Tsinghua University Spring Breeze Fund, and the Beijing Institute of Technology's Proof of Concept Project for Tumor Neoantigen Personalized TCR-T Therapy.

AUTHOR CONTRIBUTIONS

Conceptualization, M.J. and X.L.; Methodology, M.J. and Z.Y.; Validation, M.J.; Formal Analysis, M.J.; Investigation: M.J.; Data Curation, M.J.; Writing – Original Draft, M.J.; Writing – Review and Editing, M.J. and X.L.; Visualization, M.J.; Project Administration: X.L.; Supervision, X.L.; Funding Acquisition: X.L. All authors have read and approved the final manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: June 23, 2023
Revised: January 21, 2024
Accepted: April 15, 2024
Published: April 18, 2024

REFERENCES

1. Morath, A., and Schamel, W.W. (2020). αβ and γδ T cell receptors: Similar but different. J. Leukoc. Biol. *107*, 1045–1055.
2. Davis, M.M., and Bjorkman, P.J. (1988). T-cell antigen receptor genes and T-cell recognition. Nature *334*, 395–402.
3. Mora, T., and Walczak, A.M. (2016). Quantifying Lymphocyte Receptor Diversity. Preprint at bioRxiv. https://doi.org/10.48550/arXiv.1604.00487.
4. Holler, P.D., and Kranz, D.M. (2003). Quantitative Analysis of the Contribution of TCR/pepMHC Affinity and CD8 to T Cell Activation. Immunity *18*, 255–264.
5. Li, Q.J., Dinner, A.R., Qi, S., Irvine, D.J., Huppa, J.B., Davis, M.M., and Chakraborty, A.K. (2004). CD4 enhances T cell sensitivity to antigen by coordinating Lck accumulation at the immunological synapse. Nat. Immunol. *5*, 791–799.
6. Hu, Y., Wang, Z., Hu, H., Wan, F., Chen, L., Xiong, Y., Wang, X., Zhao, D., Huang, W., and Zeng, J. (2019). ACME: pan-specific peptide–MHC class I binding prediction through attention-based deep neural networks. Bioinformatics *35*, 4946–4954.
7. Jurtz, V., Paul, S., Andreatta, M., Marcatili, P., Peters, B., and Nielsen, M. (2017). NetMHCpan-4.0: Improved Peptide-MHC Class I Interaction Predictions Integrating Eluted Ligand and Peptide Binding Affinity Data. J. Immunol. *199*, 3360–3368.
8. Zeng, H., and Gifford, D.K. (2019). DeepLigand: accurate prediction of MHC class I ligands using peptide embedding. Bioinformatics *35*, i278–i283.
9. O'Donnell, T.J., Rubinsteyn, A., Bonsack, M., Riemer, A.B., Laserson, U., and Hammerbacher, J. (2018). MHCflurry: Open-Source Class I MHC Binding Affinity Prediction. Cell Syst. *7*, 129–132.e4.
10. Liu, Z., Cui, Y., Xiong, Z., Nasiri, A., Zhang, A., and Hu, J. (2019). DeepSeqPan, a novel deep convolutional neural network model for pan-specific class I HLA-peptide binding affinity prediction. Sci. Rep. *9*, 794.
11. Phloyphisut, P., Pornputtapong, N., Sriswasdi, S., and Chuangsuwanich, E. (2019). MHCSeqNet: a deep neural network model for universal MHC binding prediction. BMC Bioinf. *20*, 270.
12. Han, Y., and Kim, D. (2017). Deep convolutional neural networks for pan-specific peptide-MHC class I binding prediction. BMC Bioinf. *18*, 585.
13. Bagaev, D.V., Vroomans, R.M.A., Samir, J., Stervbo, U., Rius, C., Dolton, G., Greenshields-Watson, A., Attaf, M., Egorov, E.S., Zvyagin, I.V., et al. (2020). VDJdb in 2019: database extension, new analysis infrastructure and a T-cell receptor motif compendium. Nucleic Acids Res. *48*, D1057–D1062.
14. Vita, R., Mahajan, S., Overton, J.A., Dhanda, S.K., Martini, S., Cantrell, J.R., Wheeler, D.K., Sette, A., and Peters, B. (2019). The Immune Epitope Database (IEDB): 2018 update. Nucleic Acids Res. *47*, D339–D343.
15. Tickotsky, N., Sagiv, T., Prilusky, J., Shifrut, E., and Friedman, N. (2017). McPAS-TCR: a manually curated catalogue of pathology-associated T cell receptor sequences. Bioinformatics *33*, 2924–2929.
16. Dash, P., Fiore-Gartland, A.J., Hertz, T., Wang, G.C., Sharma, S., Souquette, A., Crawford, J.C., Clemens, E.B., Nguyen, T.H.O., Kedzierska, K., et al. (2017). Quantifiable predictive features define epitope-specific T cell receptor repertoires. Nature *547*, 89–93.
17. Glanville, J., Huang, H., Nau, A., Hatton, O., Wagar, L.E., Rubelt, F., Ji, X., Han, A., Krams, S.M., Pettus, C., et al. (2017). Identifying specificity groups in the T cell receptor repertoire. Nature *547*, 94–98.
18. Jokinen, E., Huuhtanen, J., Mustjoki, S., Heinonen, M., and Lähdesmäki, H. (2021). Predicting recognition between T cell receptors and epitopes with TCRGP. PLoS Comput. Biol. *17*, e1008814.
19. Montemurro, A., Schuster, V., Povlsen, H.R., Bentzen, A.K., Jurtz, V., Chronister, W.D., Crinklaw, A., Hadrup, S.R., Winther, O., Peters, B., et al. (2021). NetTCR-2.0 enables accurate prediction of TCR-peptide binding by using paired TCRα and β sequence data. Commun. Biol. *4*, 1–13.
20. Weber, A., Born, J., and Rodriguez Martínez, M. (2021). TITAN: T-cell receptor specificity prediction with bimodal attention networks. Bioinformatics *37*, i237–i244.
21. Moris, P., De Pauw, J., Postovskaya, A., Gielis, S., De Neuter, N., Bittremieux, W., Ogunjimi, B., Laukens, K., and Meysman, P. (2021). Current challenges for unseen-epitope TCR interaction prediction and a new perspective derived from image classification. Briefings Bioinf. *22*, bbaa318.
22. Springer, I., Besser, H., Tickotsky-Moskovitz, N., Dvorkin, S., and Louzoun, Y. (2020). Prediction of Specific TCR-Peptide Binding From Large Dictionaries of TCR-Peptide Pairs. Front. Immunol. *11*, 1803.
23. Lu, T., Zhang, Z., Zhu, J., Wang, Y., Jiang, P., Xiao, X., Bernatchez, C., Heymach, J.V., Gibbons, D.L., Wang, J., et al. (2021). Deep learning-based prediction of the T cell receptor–antigen binding specificity. Nat. Mach. Intell. *3*, 864–875.
24. Gao, Y., Gao, Y., Fan, Y., Zhu, C., Wei, Z., Zhou, C., Chuai, G., Chen, Q., Zhang, H., and Liu, Q. (2023). Pan-Peptide Meta Learning for T-cell receptor–antigen binding recognition. Nat. Mach. Intell. *5*, 236–249.
25. Meysman, P., De Neuter, N., Gielis, S., Bui Thi, D., Ogunjimi, B., and Laukens, K. (2019). On the viability of unsupervised T-cell receptor sequence clustering for epitope preference. Bioinformatics *35*, 1461–1468.
26. Petrova, G., Ferrante, A., and Gorski, J. (2012). Cross-Reactivity of T Cells and its Role in the Immune System. Crit. Rev. Immunol. *32*, 349–372.
27. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. Preprint at arXiv. https://doi.org/10.48550/arXiv.2010.11929.
28. Atchley, W.R., Zhao, J., Fernandes, A.D., and Drüke, T. (2005). Solving the protein sequence metric problem. Proc. Natl. Acad. Sci. USA *102*, 6395–6400.
29. Zhang, H., Liu, L., Zhang, J., Chen, J., Ye, J., Shukla, S., Qiao, J., Zhan, X., Chen, H., Wu, C.J., et al. (2020). Investigation of Antigen-Specific T-Cell Receptor Clusters in Human Cancers. Clin. Cancer Res. *26*, 1359–1371.
30. 10x Genomics (2022). A New Way of Exploring Immunity: Linking Highly Multiplexed Antigen Recognition to Immune Repertoire and Phenotype.
31. Hudson, D., Fernandes, R.A., Basham, M., Ogg, G., and Koohy, H. (2023). Can we predict T cell specificity with digital biology and machine learning? Nat. Rev. Immunol. *23*, 511–521.
32. Dens, C., Laukens, K., Bittremieux, W., and Meysman, P. (2023). The pitfalls of negative data bias for the T-cell epitope specificity challenge. Nat. Mach. Intell. *5*, 1060–1062.
33. Chen, S.-Y., Yue, T., Lei, Q., and Guo, A.-Y. (2021). TCRdb: a comprehensive database for T-cell receptor sequences with powerful search function. Nucleic Acids Res. *49*, D468–D474.
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., and Polosukhin, I. (2017). Attention Is All You Need. Preprint at arXiv. https://doi.org/10.48550/arXiv.1706.03762.
35. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. Preprint at arXiv. https://doi.org/10.48550/arXiv.1912.01703.
36. McKinney, W. (2010). Data Structures for Statistical Computing in Python, pp. 56–61.
37. Matplotlib: A 2D Graphics Environment (2007 (IEEE Journals & Magazine), IEEE Xplore. https://ieeexplore.ieee.org/document/4160265.
38. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.
39. Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., et al. (2020). Array programming with NumPy. Nature *585*, 357–362.
40. Wickham, H. (2009). ggplot2: Elegant Graphics for Data Analysis (Springer-Verlag).

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Code for VitTCR	This paper	Website: https://github.com/Jiang-Mengnan/VitTCR
IEDB	Vita et al.[14]	Website: https://www.iedb.org
VDJdb	Bagaev et al.[13]	Website: https://vdjdb.cdr3.net
McPAS	Tickotsky et al.[15]	Website: https://ngdc.cncb.ac.cn/databasecommons/database/id/4211
TCRdb	Chen et al.[33]	Website: https://guolab.wchscu.cn/TCRdb/#
TCR clone expansion-related data	10x Genomics.[30]	https://www.10xgenomics.com/resources/datasets/cd-8-plus-t-cells-of-healthy-donor-1-1-standard-3-0-2 https://www.10xgenomics.com/resources/datasets/cd-8-plus-t-cells-of-healthy-donor-2-1-standard-3-0-2
Software and algorithms		
pytorch	Paszke et al.[35]	RRID: SCR_018536, https://pytorch.org
pandas	McKinney et al.[36]	RRID: SCR_018214, https://pandas.pydata.org
matplotlib	J.D.Hunter et al.[37]	RRID: SCR_008624, SCR_002577 https://matplotlib.org
Scikit-learn	Pedregosa et al.[38]	RRID: SCR_002577, https://matplotlib.org
numpy	Charles et al.[39]	RRID: SCR_008633, https://numpy.org
ggplot2	Wickham et al.[40]	RRID: SCR_014601, https://ggplot2.tidyverse.org

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Xun Lan (xlan@tsinghua.edu.cn).

Materials availability

This study did not generate any new materials.

Data and code availability

- All data files for performing the analyses and generating the figures presented here are publicly available at the DOI provided in the key resources table.
- All original code has been deposited at Github and is publicly available as of the date of publication. DOIs are listed in the key resources table.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

METHOD DETAILS

Dataset processing

To train the model, we collected experimentally verified CDR3β-peptide pairs as positive samples from IEDB and McPAS. The data from VDJdb were utilized as a testing dataset to validate the model. All negative samples were generated by mismatching the peptide in each positive sample with a randomly selected CDR3β sequence from a healthy donor in TCRdb.[33] Furthermore, we focused on the CDR3β sequences with 10-20 AAs, which started with 'C' AAs and ended with 'F' or 'W' AAs, and the peptides with 8-12 AAs that were presented by human MHC class I molecules. Therefore, we obtained an HLA-A*02:01-restricted training set with 38,712 data points before cluster-based filtering and 28,584 data points after cluster-based filtering using iSMART according to our experimental needs. The number of data points in the testing set was 5,250. Additionally, TCR sequences of $CD8^+$ T cells acquired from healthy human donors (18,331 and 8,337 cells from Donor 1 and Donor 2, respectively) were used to measure the generalization of VitTCR by calculating correlations between the predicted

binding probabilities and clonal fractions. To demonstrate the reliability of VitTCR, a total of 165 experimentally validated CDR3β-peptide pairs from COVID-19 recoverees were also used to display a positive correlation between the predicted binding probabilities and the activation percentages of T cells.

Statistical analysis of sequence length

CDR3β plays a crucial role in the recognition of antigens by directly binding to antigenic epitopes. Consequently, when encoding the interactions between the TCR and antigenic epitopes, we focus solely on the CDR3β region of the TCR. Considering the specific attributes of VitTCR, it is essential to maintain a consistent input shape for the model. To determine the optimal length threshold, a thorough statistical analysis was performed on the lengths of CDR3β and antigenic epitopes within databases. As depicted in Figure S1, a substantial majority, precisely 98.89%, of the entire set of CDR3β sequences exhibit lengths spanning from 10 to 20 amino acids. Likewise, a substantial proportion, amounting to 97.97%, of all epitopes exhibited lengths within the range of 8 to 12 amino acids. Therefore, for CDR3β sequences shorter than 20 amino acids or epitopes shorter than 12 amino acids, zero padding will be applied from the N-terminal to the C-terminal end of the sequences.

Statistical analysis of HLA

We conducted a statistical analysis of HLAs across the IEDB, McPAS, and VDJdb databases, which collectively comprise a total of 104 distinct HLAs. Notably, the HLA-A*02:01 allele exhibited the highest frequency, with a count of 24,874 instances. We further examined the distribution of the top 20 HLAs with the highest frequencies within these databases (Figure S2B).

Atchleymap encoding

Prior to predicting a pair of CDR3β and antigenic epitopes, it is necessary to convert their sequence information into numerical representations. The Atchley factors consist of five factors that represent different physicochemical characteristics, with each amino acid being characterized by these five factors. For each factor, we encoded them using the principle shown in Equation 1: the subscript *i* (ranging from 1 to 20) represents the position of amino acids of CDR3βs; *j* (ranging from 1 to 12) represents the position of amino acids of epitopes; and the coordinates [i, j] correspond to the absolute difference between the values of the Atchley factor of the two amino acid residues. Each Atchley factor generates a separate map, resulting in a total of five maps. These maps are then stacked together, and the resulting output tensor is referred to as AtchleyMap.

$$\mathrm{AtchleyFactor}_{ij} = \left|\mathrm{AtchleyFactor}_{\mathrm{CDR3}\beta[i]} - \mathrm{AtchleyFactor}_{\mathrm{Epitope}[j]}\right| \quad \text{(Equation 1)}$$

Patch division

To provide a more detailed representation of the interactions, we partitioned the AtchleyMaps into distinct patches and assigned numerical labels to each patch accordingly (Figure 1B). Prior to the partitioning process, CDR3β sequences with fewer than 20 amino acids and epitopes with fewer than 12 amino acids were zero-padded from the N-terminus to the C-terminus. Consequently, we obtained 15 patches, each with a side length of 4 amino acids.

Architecture of VitTCR

As depicted in Figure 1C, VitTCR took the encoded AtchleyMaps as input and generated predicted probabilities as output. Initially, VitTCR applied a two-dimensional convolutional layer with 256 kernels (size of 4 × 4 and stride of 4) to extract informative features from AtchleyMaps. This process can also be seen as dividing the encoded AtchleyMaps into 15 patches, where each patch was vectorized into a vector of size [256] using the two-dimensional convolutional layer (size of 4 × 4 and stride of 4). After the convolution step, these vectors were concatenated into a two-dimensional vector of size [15, 256]. Subsequently, the two-dimensional vector was concatenated with a randomly generated one-dimensional vector (size of [1, 256]), which serves as a learnable classification (CLS) token. At this point, the concatenated vector, which contained 16 tokens, had a size of [16, 256]. To retain the relative positional information, a position embedding of the same size as the concatenated vector was added to the vector. Afterwards, an encoder that consisted of a normalization layer, a multi-head attention layer, a dropout layer, and a feedforwards layer was used to further encode the 16 tokens. Finally, a multilayer perceptron (MLP) layer with two units was added on top of the encoder to extract features from the CLS token, and a softmax function was applied in the final output layer to export a vector P of size [2, 1]. The shape of P is the number of classes, including "binding" and "no binding", with each element representing the probabilities related to the class labels, and the predicted probabilities summed to 1.

Model training and selection

In order to train VitTCR, we employed binary cross-entropy between the model predictions and the actual values as the loss function (Equation 2). The model was optimized using the Adam Optimizer with a learning rate of 0.001. N represents the number of samples involved in the loss function calculation. For the two categories, binding and non-binding, the probabilities predicted by VitTCR are pi and 1 - pi, with the corresponding labels being yi and 1 - yi (in this classification task, yi is 1). The hyperparameters of the model were determined through cross-validation, and the model's generalization capability was assessed using an independent test set.

$$LOSS = \frac{1}{N}\sum_{i} - \left[y_i \times \log(p_i) + (1 - y_i) \times \log(1 - p_i)\right] \quad \text{(Equation 2)}$$

Self-attention mechanism of VitTCR

The self-attention mechanism plays a crucial role in enabling VitTCR to predict the specific recognition between a CDR3β-epitope pair. As depicted in Figure 1, the Encoder module in VitTCR is similar to the Encoder module in Transformer.[34] Each token, denoted as Token_i (where i represents the token number, i ∈ [CLS,1,2...,15]), was encoded by the encoder module, obtaining a $Score_{Token\_i}$. $Score_{Token\_i}$ is a list of length 16, where each element determines the level of attention that should be assigned to the other tokens during the encoding process of Token_i. The sum of these outputed $Score_{Token\_i}$ is equal to 1 (Equation 3) (where j ∈ [CLS,1,2,...,15]). After that, the weighted average of the value vectors v_{Token_j} was calculated for all tokens based on their corresponding scores $Score_{Token\_i}[j]$, as shown in Equation 4. The resulting weighted average, denoted as $Embedded_{Token\_i}$, represents the encoded representation of the token Token_i based on the self-attention mechanism. The functioning of the self-attention mechanism is elucidated in Figure S4. Taking the encoding process of a CLS token as an illustrative example, the scores between the CLS token and the other tokens are visually represented by the thickness of the connecting lines. A thicker line signifies a higher score, while a thinner line indicates a lower score. The objective is to assign higher scores and greater weight to tokens that carry relevant information while assigning lower scores and lesser weight to tokens that contain irrelevant noise. This step is instrumental in preserving meaningful information while disregarding irrelevant noise. Although the SoftMax score between a token and itself tends to be the highest, it remains important to consider other tokens that are interconnected with the current token to capture their contextual relevance.

$$\sum_{j}^{[CLS,1,2,\ldots,15]} Score_{Token\_i}[j] = 1 \quad \text{(Equation 3)}$$

$$\text{Embedded Token\_i} = \sum_{j}^{[CLS,1,2,\ldots,15]} \left(Score_{Token\_i}[j] \times v_{Token\_j}\right) \quad \text{(Equation 4)}$$

PBWM integration

In VitTCR, the integration of PBWM involves the calculation of the mean value between $Score_{Token\_CLS}$ and PBWM (Equation 5), followed by the calculation of the embedded CLS token, as illustrated in Equation 6. To evaluate the influence of incorporating PBWM on the predictive accuracy of the model, a comparative analysis was conducted between VitTCR with the inclusion of the computed PBWM and VitTCR without the incorporation of this additional weight matrix.

$$Score_{CLS\_PBWM} = \frac{Score_{Token\_CLS}[1,15] + PBWM}{2} \quad \text{(Equation 5)}$$

$$\text{Embedded Token\_CLS} = \sum_{j}^{[CLS,1,2,\ldots,15]} \left(Score_{CLS\_PBWM}[j] \times v_{Token\_j}\right) \quad \text{(Equation 6)}$$

Cluster-based filtering

iSMART was utilized for cluster-based filtering of CDR3βs. The CDR3β sequences in the training set and the test set were filtered independently through separate clustering processes. The command line for iSMART is "python iSMARTv3.py -f cdr3β.txt -v ". During the clustering process of N CDR3β sequences, iSMART calculated the pairwise alignment score based on the BLOSUM62 matrix and normalized the score by the length of the longer CDR3β sequence. Then, an N-by-N pairwise scoring matrix was obtained. Finally, the cut-off value was set as 3.5 (default value of iSMART) to filter out all the unclustered CDR3βs.

Details of the validation of cluster-based filtering

Step 1: Assigning percentile ranks pre-filtering

Rank these epitopes based on their cognate TCR counts in descending order. The ranking is expressed in percentiles, where lower percentile values represent epitopes with higher cognate TCR counts, implying a higher frequency or prevalence of these epitopes within the dataset. Conversely, higher percentile values indicate epitopes with lower cognate TCR counts, suggesting these epitopes are less frequent. The distribution of percentile values for these epitopes is illustrated in the 'Before' category in Figure 4B, with the values representing the median.

Step 2: Examine the distribution of percentile ranks for filtered epitopes

After filtering the dataset, we plotted the percentile values of all remaining epitopes using violin plots, keeping the percentiles consistent with those obtained in the previous step. For example, before filtering, the dataset had six epitopes[A, B, C, D, E, F] with percentiles [0, 0.2, 0.4, 0.6, 0.8, 1.0]. After filtering, the dataset retained only three epitopes [A, C, E], and the percentiles for these epitopes remained [0, 0.4, 0.8]. These are shown in the 'After' category in Figure 4B.

Step 3: Comparison

Figure 4B helps illustrate changes in the distribution, including any shifts in the median and alterations in the distribution's spread (as indicated by the waist of the violin plots).

T-cell activation percentage

First, we constructed a K562 cell line expressing TMG (Tandem Minigene). TMG is a concatenated series of peptides with 11 different antigenic peptides, one of which is a wild type antigen, and the remaining ten are antigenic peptides with single amino acid mutations. Then, we used this cell line to stimulate Peripheral Blood Mononuclear Cells (PBMCs) from individuals who had recovered from COVID-19, using CD137 as a marker to sort T cells. The sorted T cells were then sequenced to identify their T Cell Receptor (TCR) sequences. We engineered T cells to express these specific TCR sequences based on the sequencing data. Subsequently, we reconstructed each of the 11 antigenic sequences from TMG, resulting in 11 different K562 cell lines, each expressing only one antigenic sequence. Finally, T cells expressing a single TCR sequence were co-cultured with K562 cells, each expressing only one type of antigenic sequence. The co-culture was analyzed by flow cytometry, using CD69 as a marker to determine activation. When calculating the activation proportion, the gating position was determined based on a negative control (non-activated group). The proportion of $CD69^{+}$ cells obtained represents the activation proportion after background removal.

QUANTIFICATION AND STATISTICAL ANALYSIS

Statistical analysis for evaluating model predictions involved calculating Spearman correlation coefficients. These analyses aimed to measure the association between VitTCR's predicted binding probabilities and observed TCR clone fractions, as well as between predicted binding probabilities and T-cell activation percentages. All statistical tests were two-sided t-tests, and results were considered statistically significant at p-values less than 0.01.