Bioinformatics, 2023, **39(12)**, btad743
https://doi.org/10.1093/bioinformatics/btad743
Advance Access Publication Date: 9 December 2023
Original Paper

OXFORD

Sequence analysis

EPIC-TRACE: predicting TCR binding to unseen epitopes using attention and contextualized embeddings

Dani Korpela [1,*], Emmi Jokinen [1,2,3], Alexandru Dumitrescu [1], Jani Huuhtanen[2,3], Satu Mustjoki[2,3,4], Harri Lähdesmäki[1,*]

[1]Department of Computer Science, Aalto University, 02150 Espoo, Finland
[2]Translational Immunology Research Program, Department of Clinical Chemistry and Hematology, University of Helsinki, 00290 Helsinki, Finland
[3]Hematology Research Unit Helsinki, Helsinki University Hospital Comprehensive Cancer Center, 00290 Helsinki, Finland
[4]iCAN Digital Precision Cancer Medicine Flagship, Helsinki, Finland

*Corresponding authors. Department of Computer Science, Aalto University, Konemiehentie 2, 02150 Espoo, Finland. E-mails: dani.korpela@aalto.fi (D.K.) and harri.lahdesmaki@aalto.fi (H.L.)

Associate Editor: Lenore Cowen

Abstract

Motivation: T cells play an essential role in adaptive immune system to fight pathogens and cancer but may also give rise to autoimmune diseases. The recognition of a peptide–MHC (pMHC) complex by a T cell receptor (TCR) is required to elicit an immune response. Many machine learning models have been developed to predict the binding, but generalizing predictions to pMHCs outside the training data remains challenging.

Results: We have developed a new machine learning model that utilizes information about the TCR from both α and β chains, epitope sequence, and MHC. Our method uses ProtBERT embeddings for the amino acid sequences of both chains and the epitope, as well as convolution and multi-head attention architectures. We show the importance of each input feature as well as the benefit of including epitopes with only a few TCRs to the training data. We evaluate our model on existing databases and show that it compares favorably against other state-of-the-art models.

Availability and implementation: https://github.com/DaniTheOrange/EPIC-TRACE.

1 Introduction

T cells are a vital part of the adaptive immune system. To determine whether an immune response is needed, T cells interact with infected, cancerous and healthy cells. Upon recognition of a target cell an immune response is elicited. This target cell recognition is based on their characterizing receptors, the T cell receptors (TCR), that bind to peptides presented by major histocompatibility complex (MHC) molecules. Thus, accurately predicting the interactions between the TCR and the peptide–MHC (pMHC) complex would be highly valuable.

The TCR consists of two chains, the α and the β chain, which both have variable regions created by somatic V(D)J-recombination. Both chains are important for the pMHC interaction and consists of three complementarity-determining regions CDR1, CDR2, and CDR3. The CDR3 is the most variable region and more in contact with the peptide, whereas the CDR1 and CDR2 regions are encoded within the V gene and are more in contact with the MHC (Rudolph and Wilson 2002). More importance has been placed on the CDR3 of the β chain than other parts of the TCR, which is also reflected in currently available TCR–pMHC data. However, the use of both chains and V and J gene information has been shown to improve the prediction accuracy (Jokinen *et al.* 2021, Moris *et al.* 2021). The V(D)J-recombination creates diversity both from a combinatorial effect by choosing which genes to include and a junctional effect stemming from random nucleotide insertions and deletions in the ligation process of the chosen gene segments. Together the two chains can form a vast TCR diversity with estimates ranging from 10^{15} to 10^{20}, being orders of magnitudes larger than the estimated amount of cells in the human body $3.7 \cdot 10^{13}$ (Laydon *et al.* 2015). Similarly as the TCRs, the pMHCs are very diverse. Naive estimates for pMHC diversity of one human are between 10^8 and 10^9 and about 10^{13} for MHC class 1 and 2, respectively (Rock *et al.* 2016). In addition to the astronomical number of possible TCR–pMHC pairs, both parts show cross-reactivity, i.e. one TCR can recognize approximately 10^6 peptides and a peptide can be recognized by many TCRs (Wooldridge *et al.* 2012).

The TCR repertoire can be studied as a whole by comparing clonalities or diversities between individuals or populations (Valkiers *et al.* 2022). The usage and evolutionary conservation of V, D, and J genes have also been studied to understand the repertoires (Valkiers *et al.* 2022). However, the underlying key concept is the TCR–pMHC binding, enabling one to understand which TCR(s) bind to which epitope(s). Many different machine learning approaches have been used to predict the TCR–pMHC binding, including clustering

Received: 30 June 2023; **Revised:** 20 November 2023; **Editorial Decision:** 29 November 2023; **Accepted:** 7 December 2023

based methods [TCRdist (Dash *et al.* 2017), GLIPH (Glanville *et al.* 2017, Huang *et al.* 2020), TCRMatch (Chronister *et al.* 2021)], decision trees [SETE (Tong *et al.* 2020)], random forests [TCRex (Gielis *et al.* 2019), epiTCR (Pham *et al.* 2023)], and Gaussian processes [TCRGP (Jokinen *et al.* 2021)]. Recently, as more data have become available, many different data-intensive deep learning approaches have been proposed [ERGO (Springer *et al.* 2020, 2021), ImRex (Moris *et al.* 2021), TITAN (Weber *et al.* 2021), NetTCR (Jurtz *et al.* 2018, Montemurro *et al.* 2021), DeepTCR (Sidhom *et al.* 2021), TCRAI (Zhang *et al.* 2021), TCRconv (Jokinen *et al.* 2023), TEINet (Jiang *et al.* 2023), TCR-BERT (Wu *et al.* 2021), PanPep (Gao *et al.* 2023), TEIM-Seq (Peng *et al.* 2023)]. Still, the complexity of the problem and the quality, amount and imbalance of the available data cause challenges for developing methods that generalize to TCRs and pMHC not included in the training data (Tong *et al.* 2020, Montemurro *et al.* 2021, Moris *et al.* 2021, Sidhom *et al.* 2021, Weber *et al.* 2021, Gao *et al.* 2023, Jiang *et al.* 2023, Pham *et al.* 2023, Peng *et al.* 2023). Moreover, most available prediction tools use epitopes only as categorical features, omitting the amino acid sequence altogether (Dash *et al.* 2017, Glanville *et al.* 2017, Gielis *et al.* 2019, Huang *et al.* 2020, Tong *et al.* 2020, Chronister *et al.* 2021, Jokinen *et al.* 2021, Sidhom *et al.* 2021, Wu *et al.* 2021, Zhang *et al.* 2021, Jokinen *et al.* 2023). This effectively leads to inability to predict binding for epitopes outside the training data. Furthermore, already limited training data have to be filtered out when training epitope-specific predictors, as there are not sufficient amounts of data per epitope. On the contrary, the ability to predict for unseen peptides would facilitate the prediction of cognate peptides to disease-associated orphan TCRs.

In this work, we present a new deep learning model, EPIC-TRACE, that utilizes ProtBERT (Elnaggar *et al.* 2021) based contextualized encodings of the amino acid sequences of the peptide and the TCR as well as multi-head attention and convolutions to achieve accurate and robust predictions. We primarily focus on predicting TCR–pMHC interactions for peptides that are not included in the training data (so-called unseen epitope task). As input to the EPIC-TRACE model we use the CDR3, V, and J genes of both chains (whenever available) and the peptide sequence together with its corresponding MHC allele. The use of protein language models for embeddings is motivated by their tendency to encode structural information which correlates well with protein function (Vig *et al.* 2021). We show that utilizing information about all available parts of the TCR–pMHC complex as input features in our model leads to best predictive performance. Furthermore, we show that including peptides that may have only a few interacting TCRs in the training data improves the performance on the unseen epitope task and demonstrate how the model can be used as an *in silico* peptide screening method. Finally, we show that our model performs better or comparable to recent models across a variety of prediction tasks.

2 Materials and methods

2.1 Data

TCR–pMHC discovery relies mostly on the use of pMHC-multimers, which are restricted to relatively few pMHCs compared to a vast amount of possible T cells screened for recognition. Thus, the current TCR–pMHC data are skewed to have far more unique TCRs than pMHCs. These skewed data make the TCR–pMHC prediction task harder. We collected our data of positive TCR–pMHC pairs from two databases: VDJdb (Bagaev *et al.* 2020) and IEDB (Mahajan *et al.* 2018).

Since both VDJdb and IEDB have much less MHC class II datapoints, we filtered the data to contain only MHC class I datapoints and further required the host to be from human. For a fair comparison we define our base dataset $\mathcal{D}_{\alpha\beta,\beta}$, which we subsample or extend as explained later in the corresponding experiments. For each datapoint in $\mathcal{D}_{\alpha\beta,\beta}$, we required the following information: the amino acid sequence of the β chain CDR3 region, the epitope amino acid sequence, and information about the βV, βJ, and MHC genes at any precision, i.e. the full-length amino acid sequence of the TCR might not be available. Dataset $\mathcal{D}_{\alpha\beta,\beta}$ contains only datapoints with sufficient β information, to which we also add information about α chain, when available. In Section 3, we use $\mathcal{D}_{\alpha\beta,\beta}$ as described above unless specified otherwise.

We unified the notation for all V and J genes and discarded datapoints with nonfunctional genes according to the IMGT (Folch and Lefranc 2000a, b, Scaviner and Lefranc 2000a, b, Lefranc *et al.* 2003). We ensured that all CDR3s are in canonical form by adding missing anchor position residues (C and F/W) and if not possible we discarded the datapoint. Datapoints that only differed in precision of gene information were filtered out by keeping only the most precise. The numbers of unique feature values of all datasets are shown in Supplementary Table S1. We note that the three most frequent epitopes, i.e. epitopes with most associated TRCs, make up more than a fourth of all datapoints in our IEDB + VDJdb based datasets.

2.2 Prediction tasks

Because of the paired nature of the data and the challenges due to the data imbalance, the TCR–pMHC prediction is more suitable to be expressed as four separate tasks. An important distinction is whether to test for epitopes contained in the training data (seen epitopes) or the converse, test for unseen epitopes. Methods treating the epitope as categorical label cannot naturally predict for unseen epitopes. This distinction is very important as it precisely defines the difficulty of the problem. Furthermore, following (Springer *et al.* 2020) the tasks can similarly be divided in terms of seen or unseen TCRs, resulting in the following three tasks: TCR–Peptide Pairing 1 (TPP1) where both TCR and epitope parts of a test datapoint are seen in training data but in different pairs, TPP2 where the epitope is seen in training but TCR is unseen, and TPP3 where neither the TCR nor the epitope is seen in training. To complete the task definitions we add TPP4, where the TCR is seen in training data but the epitope is unseen. The tasks are illustrated in Supplementary Fig. S4.

The different tasks correspond to different biological questions. TPP2 seeks to answer if a TCR repertoire has T cells targeting given epitope(s), e.g. SARS-COV-2 or HIV, such that we have data on those epitopes in training. In the case our training data contain neither the epitopes nor the TCRs, the task changes to TPP3, which is arguably the most general and interesting task. Even though all tasks are relevant, currently only the two first tasks (TPP1 and TPP2) can be solved with reasonable performance. However, as individuals have generally very little overlap in their TCR

repertoires the unseen TCR tasks (TPP2 and TPP3) are more generally applicable. In addition, TCRs in the current databases, such as IEDB and VDJdb, are mostly specific to a single epitope, i.e. the TCRs appear as a pair to only one epitope. This means that the amount of positive datapoints for the TPP4 task is very low and makes it unfeasible to test with. Thus, we focus our experiments primarily to the TPP3 task, but we also include TPP2 experiments as a comparison. For the TPP4 evaluation we use an external dataset that we describe later.

2.3 Cross-validation and performance metrics

Following the common practice in the field, the performance of our model was evaluated using 10-fold cross-validation that was repeated five times. In each cross-validation fold we split the data to train and test sets and extract part of the train set for validation used for early stopping. We report the performance measures as the mean and standard error of the mean across the five cross-validation runs. We use the area under the receiver operating characteristics (AUROC) and the average precision (AP) metrics.

10x Genomics (2020), but these are not generally available for all epitopes. The generation of negative data by shuffling the positive datapoints is established in the field and gives a more reliable estimate of model performance compared to usage of external TCR datasets (Moris *et al.* 2021). The negatives are generated separately for every train (+ validation) and test set in each cross-validation fold to ensure a larger amount of negatives to shuffle epitopes in train. Importantly, this also restricts data leakage from test to train for the corresponding task. We generated the negative data by shuffling TCRs (CDR3, V, and J for both chains) with epitopes (epitope and MHC) such that the new datapoint was not in the set of positive datapoints. The randomly generated datapoint was determined negative if any part (CDR3, V, or J gene) of either chain was different to the positive TCRs of the epitope. As epiTRC, TITAN, and ImRex only consider the β chain and the epitope, the above definition can create some CDR3β-Epitope pairs that have both positive and negative labels. To ensure a fair comparison with epiTCR, TITAN, and ImRex, we created a second version of cross-validation splits of $\mathcal{D}_{\alpha\beta,\beta}$, where we determined the negatives such that at least the CDR3β had to differ. In both settings we did not allow duplicate datapoints. We created five times as many negatives as positives for each epitope as in (Montemurro *et al.* 2021, Meysman *et al.* 2023). This means that not only has the test set the ratio of 1:5 but also any individual epitope. For most frequent epitopes, there are not enough TCRs to create enough negatives by shuffling. In these cases, we discarded positive datapoints randomly to maintain the correct ratio. In order to comply with the given TPP task definitions, the splits were created separately for each task.20

As the majority of the datapoints belong to a small amount of most frequent epitopes, we balanced the amount of epitopes and datapoints in each fold for TPP3. More specifically, we ordered the epitopes in descending frequency order and then randomly assigned a fold index for $k = 10$ consecutive epitopes at a time. Importantly, this also assures that all folds have both frequent and less frequent epitopes. The TCRs are more evenly distributed and thus the cross-validation folds for TPP2 can be done simply by choosing TCRs to the splits. If any of the epitopes in the test set is not present in train, extra negatives were added to the train to obtain the TPP2 constraint (and naturally the 1:5 ratio cannot be retained for those epitopes).

2.4 EPIC-TRACE model

Our model utilizes the full TCR–pMHC information available and is designed to predict interaction between a TCR and an pMHC, i.e. a binary classification problem.

TCR features. A TCR is defined as $\text{TCR} = (\beta_\text{V}, \beta_\text{J}, \beta_\text{CDR3}, \alpha_\text{V}, \alpha_\text{J}, \alpha_\text{CDR3})$, where $\beta_\text{V} \in \mathbb{B}^{g_{\beta\text{V}}}$ and $\beta_\text{J} \in \mathbb{B}^{g_{\beta\text{J}}}$ are one-hot encoded vectors indicating the V and J genes in the β chain, $\mathbb{B} = \{0, 1\}$, and $g_{\beta\text{V}}$ and $g_{\beta\text{J}}$ denote the numbers of V and J genes (similarly for the α chain, $\alpha_\text{V} \in \mathbb{B}^{g_{\alpha\text{V}}}$ and $\alpha_\text{J} \in \mathbb{B}^{g_{\alpha\text{J}}}$). Variable $\beta_\text{CDR3} \in \mathbb{R}^{l_\beta \times e}$ consists of two parts: (i) contextualized information about the CDR3 region of the β chain that is obtained from the pre-trained ProtBERT language model (size $l_\beta \times 1024$) (Elnaggar *et al.* 2021), and (ii) one-hot encoded CDR3 region. These are concatenated to form feature representation of size $l_\beta \times e$, where l_β denotes the length of the CDR3 region that is further padded to CDR3 maximum length $l \times e$. If the V and J gene information is available for the β chain, the full-length TCR β amino acid sequence is constructed and embedded with ProtBERT. For full-length TCRs only the CDR3 region positions are extracted from the ProtBERT embedding and stored in β_CDR3. This is done as we use the V and J genes as separate inputs, and as shown by Jokinen *et al.* (2023) the contextualized CDR3 captures the essential features for classification. If the full TCR cannot be constructed, only the CDR3 region is embedded with ProtBERT and no further extraction is done. If the α chain is available, $\alpha_\text{CDR3} \in \mathbb{R}^{l_\alpha \times e}$ is defined similarly.

Epitope–MHC features. The epitope–MHC complex is defined as $\text{pMHC} = (\text{Epitope}, \text{MHC})$, where $\text{Epitope} \in \mathbb{R}^{l_e \times e}$ is obtained by concatenating the ProtBERT embedding and the one-hot encoding of the epitope sequence (and subsequent padding to maximum length), l_e is the length of epitopes, $\text{MHC} \in \mathbb{B}^{g_m}$ is the one-hot encoded vector of the MHC allele, and g_m is the number of alleles.

Output labels. We formulate our model using three separate binary output labels $y = (y_\alpha, y_\beta, y_{\alpha\beta})$, where $y_\alpha \in \mathbb{B}$, $y_\beta \in \mathbb{B}$, and $y_{\alpha\beta} \in \mathbb{B}$. If only the β chain is available, then y_α and $y_{\alpha\beta}$ are considered as missing (similarly if only y_α is available). If both α and β chains are available, then $y_{\alpha\beta}$ defines the binding and y_α and y_β are considered missing. The prediction problem is then defined with datapoints $(\text{TCR}_n, \text{pMHC}_n, y_n)$, where $n \in \{1, \ldots, N\}$, and N is the number of positive and negative datapoints.

Architecture. Our architecture utilizes convolutions, multi-head self-attentions, learnable linear embeddings and ReLU activations. The model contains three output heads corresponding to the cases when only β, only α, or both β and α chains are available. An overview of the model is shown in Fig. 1. The representations of the CDR3 regions (β_CDR3 and α_CDR3) and the epitope (Epitope) are first processed with 1-D convolutions to infer binding motifs from either the ProtBERT embedding or the one-hot encodings. Epitope convolution is concatenated separately with the CDR3 convolution of the β and α chains (if available). Multi-head attention is used to identify the important interacting features separately for $(\beta_\text{CDR3}, \text{Epitope})$ and $(\alpha_\text{CDR3}, \text{Epitope})$ pairs. In this way the model can handle missing information in either of the chains and, importantly, the model will benefit from data with a missing chain even if the test data points would have

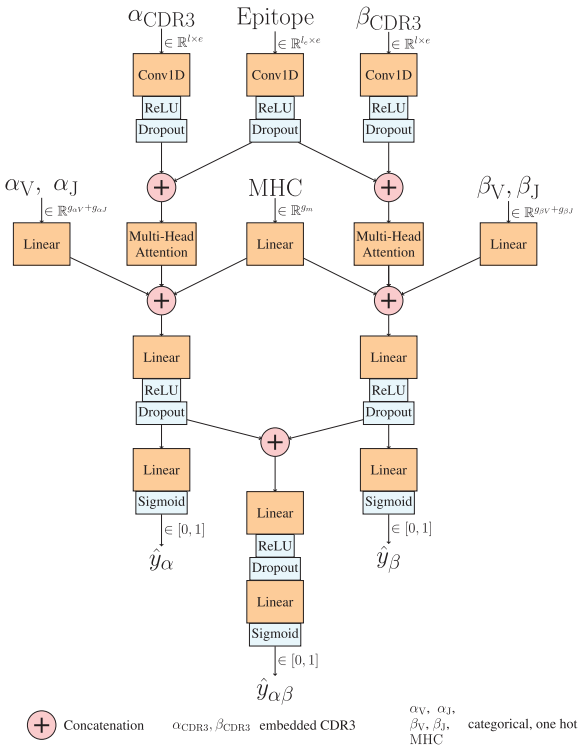

Figure 1. Architecture of the EPIC-TRACE model.

both chains (the same neural network parameters for α and β chains are respectively used in either missing or full data use cases). Learnable linear embedding is trained for the one-hot encoded V and J genes from both chains $(\beta_{\text{V}}, \beta_{\text{J}}, \alpha_{\text{V}}, \alpha_{\text{J}})$ as well as for the MHC allele (MHC), which are then concatenated with the outputs of the attentions. The β and α chains are processed with the multilayer perceptrons (linear and ReLu) separately as well as together (whenever both chains are available) and passed through sigmoidal activation to make the predictions $\hat{y}_\beta \in [0, 1]$, $\hat{y}_\alpha \in [0, 1]$ and $\hat{y}_{\alpha\beta} \in [0, 1]$ corresponding to the three different cases. Details of the neural network architecture are shown in Supplementary Section S2.

Model training. We trained our model by maximizing the logarithm of the Bernoulli likelihood or equivalently the negative binary cross-entropy

$$\text{BCEL}(\theta) = -\frac{1}{N}\sum_{n=1}^{N} w_n \Big(y_n \log(\hat{y}_n) + (1 - y_n) \log(1 - \hat{y}_n) \Big),$$

where w_n is the weight for the nth datapoint. We weighted the positive datapoints five times higher than the negatives. We controlled for over-fitting by using early stopping based on the average precision on the validation set, and the model parameters giving the highest validation score were used. Following the main training we used stochastic weight averaging (SWA) (Izmailov *et al.* 2018) for 20 epochs. We used different learning rates for training models for the TPP2 and TPP3 tasks in both the main training and the SWA sampling, 0.0001 and 0.001, respectively. In addition, we used exponential learning rate scheduler for the main training for TPP3.

3 Results

3.1 Choice of validation set, and per epitope scores

We first set out to investigate the effect of the validation set (used for early stopping) on the test performance. We compared two different ways to generate the validation set: (i) naive random sample of datapoints from the train set, and (ii) creating unseen epitope validation by choosing datapoints by epitopes from the train set. The comparison was made only for the TPP3 task, where epitope is unseen, as the unseen epitope validation is not sensible for seen epitope tasks TPP1 or TPP2. The random validation set had slightly better performance compared to the unseen epitope validation (see Supplementary Table S4), perhaps because that leaves more distinct epitopes in the training set. The unseen epitope relation is present between both train-validation (TPP3 or TPP4) and train-test (TPP3). However, the epitope distributions in validation and test are naturally distinct for the TPP3 task. Due to this inherent epitope covariate shift (as a result of very few epitopes in the current data) a representative validation set is hard to construct. Because the random validation is better representing the other tasks and also resulted in slightly better performance for the TPP3, we chose to use the random validation in all following experiments.

Due to the highly imbalanced data the joint prediction accuracy measures (AUROC and AP) are dominated by the epitopes with most datapoints. Therefore, we quantified the per epitope scores for the TPP2 and TPP3 tasks. We observe that the epitopes with more datapoints have a higher score on average on the TPP2 task (Fig. 2 left), which is logical as there are more datapoints for those epitopes to train on. To better characterize the trend explained by the number of datapoints for an epitope in the TPP2 task, we binned the per epitope scores and calculated the bin averages (Supplementary Fig. S1). The AUROC scores seem to slightly increase as the number of datapoints increases. On the other hand, we observe that the number of datapoints per epitope does not affect the performance on the TPP3 task as expected (Fig. 2 right), since by the TPP3 definition the datapoints for a specific epitope are not included in the training and, thus, only affect the number of test datapoints. Overall, prediction accuracies vary across epitopes, which can be due to the currently available data for that epitope or underlying biophysical reasons.

Furthermore, we investigated the effect of a distance between epitopes in the training and test sets. This was done by quantifying the minimum (Levenshtein) edit distance between an epitope in the test set and the epitopes in the training set. Similarly as in Moris *et al.* (2021), we observe that the per epitope scores seems to slightly decrease when the minimum edit distance to the training set increases (Fig. 3a). To further investigate the generalization performance on diverse epitope sequences, we carried out an additional experiment where we stratified the training and test folds according to minimum edit distances between the folds. More specifically, we required at least a distance of five between any two epitopes that belong to two different folds, leading to a minimum distance of five between training and test sets. The scores (AUROC 0.693 and AP 0.288) are similar to the scores obtained from the unrestricted cross-validation (see Row 7 in Table 1). These analyses suggest that our proposed model can generalize to data points outside the training data.

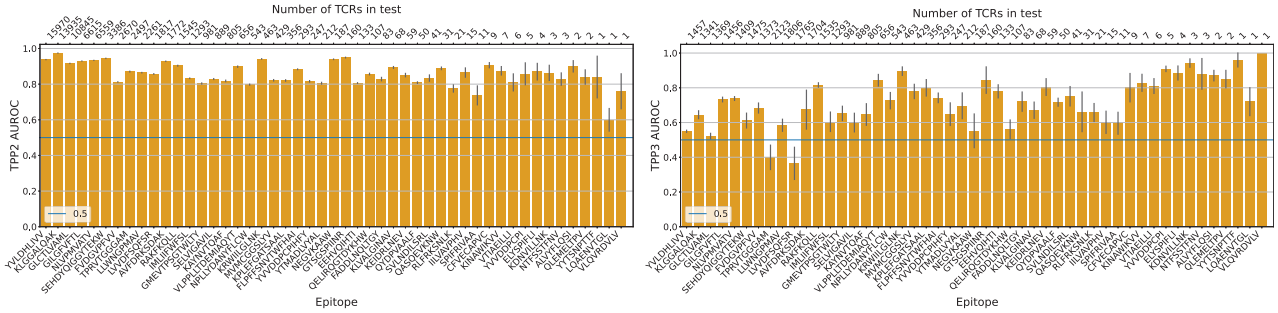

Figure 2. Per epitope AUROC values for the TPP2 (left) and TPP3 (right) tasks. Epitopes were sampled logarithmically to include epitopes with varying number of TCRs. Top *x*-axis shows the number of positive datapoints for each epitope (bottom *x*-axis). The vertical axis shows the mean of five 10-fold cross-validations runs together with the standard error.

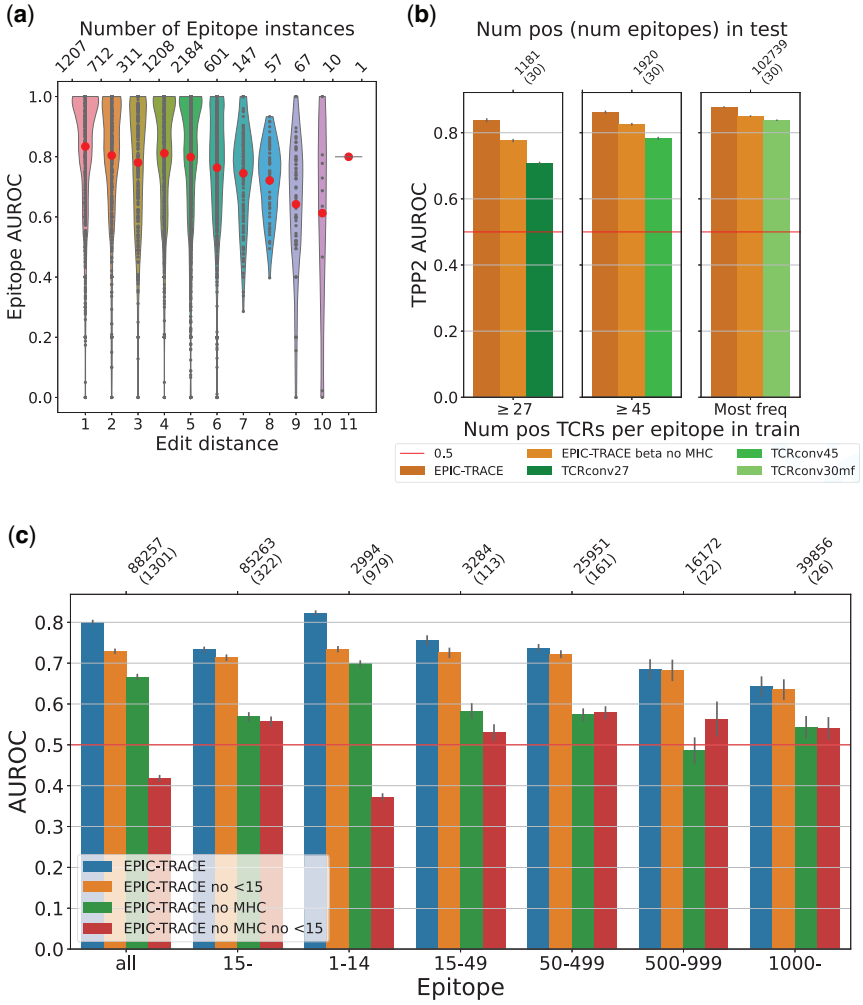

Figure 3. (a) Violin plot of AUROC scores grouped by minimum edit distance to train dataset. The large (red) dots are (unweighted) averages of the scores for the given minimum edit distance. (b) Comparison to TCRconv. TCRconv was trained on three subsets of 30 epitopes from the $\mathcal{D}_{\alpha\beta,\beta}$ dataset and compared to EPIC-TRACE trained on full $\mathcal{D}_{\alpha\beta,\beta}$ folds either with all or reduced input features. The *y*-axis shows average per epitope AUROC values of frequency binned epitopes with standard error. (c) Comparison of models trained with all datapoints or by discarding epitopes with <15 TCRs from training for TPP3. Models were trained with or without MHC information. The *y*-axis shows the average per epitope AUROC with standard error.

3.2 Input feature contribution

To study which input features are important we conducted an ablation study and trained the model using different features. We studied the performance gain of using the full length TCRs (long context) when possible for creating ProtBERT embeddings from which the CDR3 part is extracted, compared to using always only the CDR3 region as input for the ProtBERT model. In addition, we trained the models with or without the categorical V, J, and MHC information. Models were trained separately for TPP2 and TPP3 tasks. The results are presented in Table 1 and discussed below.

Interestingly, the two tasks benefited in different magnitudes of the different features. The VJ gene information given either as categorical features or as part of the context to the

Table 1. Effect of input features.[a]

	TPP2 AUROC	TPP2 AP	TPP3 AUROC	TPP3 AP
1. $\alpha\beta$ (CDR3)	0.830 ± 0.000	0.574 ± 0.000	0.513 ± 0.008	0.179 ± 0.003
2. $\alpha\beta$ (CDR3) + VJ	0.891 ± 0.000	0.665 ± 0.001	0.548 ± 0.007	0.192 ± 0.004
3. $\alpha\beta$ (CDR3) + MHC	0.837 ± 0.000	0.583 ± 0.000	0.611 ± 0.002	0.243 ± 0.002
4. $\alpha\beta$ (CDR3) + VJ + MHC	0.897 ± 0.000	0.676 ± 0.000	0.692 ± 0.007	0.289 ± 0.006
5. $\alpha\beta$ (long)	0.888 ± 0.000	0.663 ± 0.000	0.528 ± 0.008	0.191 ± 0.004
6. $\alpha\beta$ (long) + MHC	0.893 ± 0.000	0.674 ± 0.000	0.682 ± 0.010	0.284 ± 0.007
7. $\alpha\beta$ (long) + VJ + MHC	**0.906** ± 0.000	**0.698** ± 0.000	0.691 ± 0.008	0.291 ± 0.005
8. $\alpha\beta$ (long) + VJ + MHC [$\mathcal{D}_{\alpha\beta,\alpha,\beta}$]	**0.906** ± 0.000	0.691 ± 0.001	**0.693** ± 0.008	**0.294** ± 0.007

[a] The model was trained on $\mathcal{D}_{\alpha\beta,\beta}$ using different subsets of the input features. Here CDR3 and long in parenthesis denote the context used for the ProtBERT embeddings and VJ and MHC denote if the respective categorical features were used. We also compared the model on $\mathcal{D}_{\alpha\beta,\alpha,\beta}$ that contains also datapoints that have only the α chain but not the β chain (row 8). Reported values are the mean of the five 10-fold cross-validation runs together with the standard error. The values corresponding to best performing configurations are bolded.

Table 2. Comparison of TCR chains.[a]

Used chain(s)	TPP2 AUROC	TPP2 AP	TPP3 AUROC	TPP3 AP
$\alpha\beta$	0.767 ± 0.001	0.505 ± 0.001	0.541 ± 0.008	0.204 ± 0.006
β	0.721 ± 0.001	0.441 ± 0.001	0.539 ± 0.004	0.202 ± 0.005
α	0.725 ± 0.001	0.442 ± 0.001	0.537 ± 0.005	0.206 ± 0.004

[a] The model was trained with either or both of the TCR chains on a more stringent dataset $\mathcal{D}_{\alpha\beta}$, where each datapoint contains both chains. Reported values are the mean of the five 10-fold cross-validation runs together with the standard error.

ProtBERT embeddings was more important for the TPP2 task, while the MHC information was more important for the TPP3 task (Table 1). Even though the vast majority of the datapoints had either "HLA class 1" or "HLA*02:01" as their MHC information the MHC feature showed to be important. For the TPP3 task, the performance without the MHC information is much lower than when using it, even if VJ information is used. The results are logical when comparing the MHC importance between the tasks. In the TPP3 case the MHC information can be included in the training data, and thus some information about the pMHC complex can be directly used in test predictions. On the other hand, in the TPP2 task, most of the datapoints for one epitope share the same MHC information and thus this information becomes redundant, which explains the lower improvement. Expectedly, both tasks had best performance when using both VJ and MHC information. When using both VJ and MHC information the gene-gene preferences can be explicitly modeled, which could explain synergistic improvement on the TPP3 task.

To investigate the importance of the TCR chains, we evaluated the EPIC-TRACE model on a reduced dataset ($\mathcal{D}_{\alpha\beta}$), where every datapoint has necessary information of both chains available. With $\mathcal{D}_{\alpha\beta}$ we required that TCRs in the test sets differed on both α and β CDR3s from TCRs in the train. Similarly, we required both CDR3s to differ when determining negative pairs. We trained our model utilizing only either chain and with both chains. Results are shown in Table 2. On TPP2 task, using both chains outperforms the models that were trained on only the α or β chain, whereas the performances are similar on the TPP3 task. However, when using only either chain the performances are very similar, for both tasks. We note that the performances on the reduced dataset $\mathcal{D}_{\alpha\beta}$ are worse than on the full dataset $\mathcal{D}_{\alpha\beta,\beta}$ due to a smaller sample size.

Lastly, we combined our base dataset (i.e. $\mathcal{D}_{\alpha\beta,\beta}$ that contains both $\alpha\beta$ and β datapoints) with datapoints containing only the α chain (i.e. $\mathcal{D}_{\alpha\beta,\alpha,\beta}$). This could not be done with the other models that are compared in this paper as they require β chain (ERGO-II) or can only utilize either of the chains (epiTCR, TITAN, and ImRex). The combination was done by adding the new datapoints to the training sets leaving the test sets the same and comparable. Adding the α datapoints increased the TPP3 performance but lowered the AP on the TPP2, see row 8 Table 1

3.3 Increasing the amount of unique epitopes improves generalization

To investigate how the number of unique epitopes in the training data affects the two tasks (TPP2 and TPP3), we evaluated the model with two settings: (i) we included all epitopes in the cross-validation (i.e. the same standard cross-validation as above), and (ii) we discarded the epitopes with <15 TCRs from training. These settings were also extended to test sets such that the test set either included or excluded the less frequent epitopes. These low frequency epitopes comprise approximately 75% of the (1301) epitopes but only 2994 of the 147 346 datapoints. In earlier work low frequency epitopes have been discarded from the data: e.g. epitopes with <15 TCRs were excluded in TITAN (Weber *et al.* 2021), and epitopes with <10 were excluded in TEInet (Jiang *et al.* 2023). The performance scores for the two tasks and the two different settings are shown in Supplementary Table S5. When testing on all epitopes, we observe an apparent increase in the performance for the TPP3 task when the low frequency epitopes are included in the training data (AP increases from 0.280±0.008 to 0.291±0.005). Interestingly, there is only little to no improvement when testing on only more frequent epitopes in TPP3 (AP increases from 0.278±0.006 to 0.285±0.005). The TPP2 task scores did not improve with the added epitopes.

To further investigate the effect of low frequency epitopes on the low frequency and the more frequent epitopes separately, we calculated the average per epitope scores for the

different settings. Figure 3c shows that including the low frequency epitopes in training improves the results on the TPP3 task. This is especially apparent for the low frequency epitopes in the test set. Overall, the result in Fig. 3c shows that utilizing the low frequency epitopes in training is beneficial for generalization. We note that the low frequency epitopes are associated to many HLA alleles that are not present in the data of the more frequent epitopes. Since the MHC information improves the results on the TPP3 task as shown in Section 3.2, we wanted to confirm that it is indeed the addition of different epitope sequences that improves the result, not just the addition of MHC alleles. To confirm that, we trained our model without the MHC information in the same two settings. Figure 3c shows that including low frequency epitopes in the training data results in a similar performance improvement even when the EPIC-TRACE model is trained without the MHC information, thus supporting our hypothesis.

3.4 Comparisons to other methods

Next, we compared our method to other state of the art models that treat the epitope as an amino acid sequence. We compared against ERGO-II (Springer *et al.* 2020, 2021), TITAN (Weber *et al.* 2021), ImRex (Moris *et al.* 2021), and epiTCR (Pham *et al.* 2023). ERGO-II uses LSTMs to embed the CDR3 (β or $\alpha\beta$) and epitope sequences in addition to V, J MHC and T cell type class labels. ImRex utilizes a matrix of pairwise physicochemical features between CDR3 (β or α) and the epitope sequence as an input to a convolutional neural network. TITAN uses convolutions and context attention to make the prediction from the SMILES embedded epitope and BLOSUM62 embedded full length TCR (β or α). Importantly TITAN is also pretrained on a more general protein ligand binding task using SMILES. epiTCR uses random forest to predict the BLOSUM62 embedded CDR3 (β) and epitope also utilizing a 34-amino acid-long pseudosequence for the HLA. We used again the dataset $\mathcal{D}_{\alpha\beta,\beta}$ and exactly the same cross-validations data splits for all methods. The results in Table 3 show that our model outperforms epiTCR, TITAN and ImRex by a large margin, and performs consistently better than ERGO-II on both tasks. One reason to the difference can be that epiTCR, TITAN, and ImRex only utilize the β chain and the β-CDR3, respectively, compared to our model and ERGO-II utilizing all available information. Performance of all models remained consistent when using the different definitions for negative datapoints (see Supplementary Table S2).

We additionally assess the generalization performance on unseen epitopes from independent test data. For this we collected all recently added data points from the IEDB and VDJDB databases, i.e. all experimentally measured TCR–epitope–MHC interactions that were added to either IEDB or VDJDB after extraction of the $\mathcal{D}_{\alpha\beta,\alpha,\beta}$ dataset that we have used. We restricted the new test data points to have both distinct epitopes and distinct CDR3β sequences from those in the train data, i.e. the new data points belong to the TPP3 task for the previous training train $\mathcal{D}_{\alpha\beta,\alpha,\beta}$. The negatives for the new test data points were generated similarly as for train in a ratio 1:5 per epitope, where unseen TCRs were randomly chosen for each epitope. Altogether, the new independent dataset contains 2400 positive and negative data points. EPIC-TRACE compared favorably against the other methods based on the average per epitope AUROC (see Supplementary Fig. S3).

We also compared our model against a state of the art model that uses epitopes as class labels, TCRconv (Jokinen *et al.* 2023). Since the number of unique epitopes in the dataset originally used for TCRconv is in the order of tens, we trained TCRconv separately with three subsets of 30 epitopes from the $\mathcal{D}_{\alpha\beta,\beta}$, stratified according to the number of TCRs per epitope in the train set (i.e. epitopes with $\geq$27, $\geq$45, or $\geq$780 TCRs in the train set, the last one presenting the most frequent epitopes). This was done for a more fair comparison as opposed to using hundreds of epitopes. For a more detailed description of the comparison see Supplementary Section S1. Figure 3b shows that EPIC-TRACE performs better on all three subsets with both the full model and the reduced model (only β chain and no MHC). As expected, the more frequent epitopes receive a better mean AUROC score than the less frequent epitopes for both EPIC-TRACE and TCRconv. Importantly, the difference between TCRconv and EPIC-TRACE increases when the epitope frequency decreases, showcasing the advantage of using the epitope amino acid sequence. We also tested EPIC-TRACE against TCRconv on the most abundant epitopes using both α and β sequences. This is a setting where methods that treat epitopes as class labels are strongest. We observed that TCRconv can achieve a comparable performance in this setting (see Supplementary Table S3), but as discussed above, TCRconv or other similar tools cannot make prediction for any other epitopes than those in the training data.

3.5 Prediction of yeast display data

Next, we demonstrate how EPIC-TRACE can be used to screen epitopes for disease-associated TCRs—a computational task that is notoriously difficult but would have tremendous potential e.g. in understanding disease pathogenesis. Recently, Yang *et al.* (2022) identified five orphan TCRs that are associated with ankylosing spondylitis (AS) as well as acute anterior uveitis (AAU) and used yeast display library screening followed by subsequent validation to identify 26 HLA-B*27:05 restricted shared self-peptides and microbial

Table 3. Comparison to previous methods.[a]

	TPP2 AUROC	TPP2 AP	TPP3 AUROC	TPP3 AP
EPIC-TRACE (our) [$\mathcal{D}_{\alpha\beta,\alpha,\beta}$]	**0.906** ± 0.000	0.691 ± 0.001	**0.693** ± 0.008	**0.294** ± 0.007
EPIC-TRACE (our)	**0.906** ± 0.000	**0.698** ± 0.000	0.691 ± 0.008	0.291 ± 0.005
ERGO-II	0.895 ± 0.002	0.659 ± 0.007	0.675 ± 0.007	0.274 ± 0.004
epiTCR	0.793 ± 0.000	0.581 ± 0.000	0.515 ± 0.001	0.183 ± 0.001
TITAN	0.786	0.454	0.577	0.204
ImRex	0.697	0.420	0.519	0.178

[a] EPIC-TRACE, ERGO-II, and epiTCR are evaluated on five 10-fold cross-validation runs, whereas TITAN and ImRex are evaluated on only one of the five cross-validations due to long training time. Reported values are the mean of the five 10-fold cross-validation runs together with the standard error. Values for best performing models are bolded.

peptides that activated the five AS- and AAU-derived TCRs. Here we demonstrate that machine learning methods are starting to reach sufficient accuracy to complement, and eventually replace, the laborious yeast display library screening. We used the five experimentally validated TCRs and 26 epitopes, altogether 81 HLA-B*27:05 restricted TCR–peptide pairs, as positive datapoints, and created negative datapoints by assigning 2000 randomly selected HLA-B*27:05 restricted epitopes from IEDB to the five TCRs. EPIC-TRACE model trained with VDJDB + IEDB dataset $\mathcal{D}_{\alpha\beta,\alpha,\beta}$ performed poorly on the yeast display dataset (AUROC 0.485). This was also the case for ERGO-II (AUROC 0.195). This prediction task is challenging because all the datapoints have the same HLA and V alleles, meaning that the distinction has to be made purely by peptide and CDR3 sequences. Therefore, we included randomly chosen 1–4 distinct epitopes corresponding to 2–14 positive data points into the train set. For each yeast display peptide included into the training set, we generated negatives by pairing this peptide to random TCRs from the original training set to obtain the ratio 1:5, leaving the 2000 HLA-B*27:0 restricted negatives only for testing. The procedure was repeated 10 times such that all positive yeast display data points were added to train once, while evaluating on the rest of the data (unseen epitope, TPP4/TPP3). The average AUROC and AP scores were 0.807 and 0.303, respectively. The recall and number of true positives against the number of best scoring test data points are shown separately for each part in Supplementary Fig. S2. We note that all individual AUROC values are above 0.5 and from the 50 highest prediction values 20 are positive on average. This analysis shows that by utilizing approximately as little as 10%, or on average 8 positive datapoints, the performance of the model in the yeast display library task is at least moderately good.

4 Discussion

Here, we have presented EPIC-TRACE, a novel method for predicting TCR–pMHC binding using the full TCR information together with the peptide amino acid sequence and MHC allele. We showed that the seen and unseen epitope tasks behave differently and have different importance for the used input features. It is apparent that current data mostly obtained with the use of pMHC-multimers are very imbalanced and lead to difficulties to generalize to the full TCR–pMHC space. More specifically, the unseen epitope task remains very hard for state-of-the-art methods. We showed that specificity to some epitopes is easier to predict than to others, which results in varying predictive performance across epitopes. Although the simple minimum edit distance to train set in the TPP3 case explained the general difficulty, it is not accurate enough to be used as an estimate for prediction accuracy for any specific epitope. An estimate of the reliability of the prediction would be very useful for both the seen and unseen tasks. Furthermore, the development and use of new TCR–pMHC sequencing methods increase the throughput and quality of the data. Especially important is that the amount of distinct epitopes increases, even if these epitopes are not associated to many TCRs, thus also the unseen epitope task becomes more feasible to solve.

Acknowledgements

We acknowledge the computational resources provided by the Aalto Science-IT Project.

Supplementary data

Supplementary data are available at *Bioinformatics* online.

Conflict of interest

None declared.

Funding

This work was supported by the Academy of Finland; the Sigrid Juselius Foundation; and Cancer Foundation Finland.

Data availability

Data are obtained from public databases VDJDB (Bagaev *et al.* 2020) and IEDB (Mahajan *et al.* 2018), Preprocessed versions and $\mathcal{D}_{\alpha\beta,\beta}$ cross-validation splits and $\mathcal{D}_{\alpha\beta,\alpha,\beta}$ full data available, see https://github.com/DaniTheOrange/EPIC-TRACE.

References

10x Genomics. A new way of exploring immunity-linking highly multiplexed antigen recognition to immune repertoire and phenotype. *Technol Network*, 2020.

Bagaev DV, Vroomans RMA, Samir J *et al.* VDJdb in 2019: database extension, new analysis infrastructure and a T-cell receptor motif compendium. *Nucleic Acids Res* 2020;**48**:D1057–62.

Chronister WD, Crinklaw A, Mahajan S *et al.* TCRMatch: predicting T-cell receptor specificity based on sequence similarity to previously characterized receptors. *Front Immunol* 2021;**12**:640725.

Dash P, Fiore-Gartland AJ, Hertz T *et al.* Quantifiable predictive features define epitope-specific T cell receptor repertoires. *Nature* 2017;**547**:89–93.

Elnaggar A, Heinzinger M, Dallago C *et al.* ProtTrans: towards cracking the language of lifes code through self-supervised deep learning and high performance computing. *IEEE Trans Pattern Anal Mach Intell* 2021;44:7012–27.

Folch G, Lefranc M-P. The human T cell receptor beta diversity (TRBD) and beta joining (TRBJ) genes. *Exp Clin Immunogenet* 2000a;**17**:107–14.

Folch G, Lefranc M-P. The human T cell receptor beta variable (TRBV) genes. *Exp Clin Immunogenet* 2000b;**17**:42–54.

Gao Y, Gao Y, Fan Y *et al.* Pan-peptide meta learning for T-cell receptor–antigen binding recognition. *Nat Mach Intell* 2023;**5**:236–49.

Gielis S, Moris P, Bittremieux W *et al.* Detection of enriched T cell epitope specificity in full T cell receptor sequence repertoires. *Front Immunol* 2019;**10**:2820.

Glanville J, Huang H, Nau A *et al.* Identifying specificity groups in the T cell receptor repertoire. *Nature* 2017;**547**:94–8.

Huang H, Wang C, Rubelt F *et al.* Analyzing the *Mycobacterium tuberculosis* immune response by T-cell receptor clustering with GLIPH2 and genome-wide antigen screening. *Nat Biotechnol* 2020;**38**:1194–202.

Izmailov P, Podoprikhin D, Garipov T *et al.* Averaging weights leads to wider optima and better generalization. In: *Conference on Uncertainty in Artificial Intelligence*, 2018. https://doi.org/10.48550/arXiv.1803.05407.

Jiang Y, Huo M, Cheng Li S *et al.* TEINet: a deep learning framework for prediction of TCR–epitope binding specificity. *Brief Bioinform* 2023;**24**:bbad086.

Jokinen E, Huuhtanen J, Mustjoki S *et al.* Predicting recognition between T cell receptors and epitopes with TCRGP. *PLoS Comput Biol* 2021;**17**:e1008814.

Jokinen E, Dumitrescu A, Huuhtanen J *et al.* TCRconv: predicting recognition between T cell receptors and epitopes using contextualized motifs. *Bioinformatics* 2023;**39**:btac788.

Jurtz VI, Jessen LE, Bentzen AK *et al.* NetTCR: sequence-based prediction of TCR binding to peptide-MHC complexes using convolutional neural networks. *bioRxiv* 2018, https://doi.org/10.1101/433706.

Laydon DJ, Bangham CRM, Asquith B *et al.* Estimating T-cell repertoire diversity: limitations of classical estimators and a new approach. *Phil Trans R Soc B* 2015;**370**:20140291.

Lefranc M-P, Pommié C, Ruiz M *et al.* IMGT unique numbering for immunoglobulin and T cell receptor variable domains and Ig superfamily V-like domains. *Dev Comp Immunol* 2003;**27**:55–77.

Mahajan S, Vita R, Shackelford D *et al.* Epitope specific antibodies and T cell receptors in the immune epitope database. *Front Immunol* 2018;**9**:2688. page

Meysman P, Barton J, Bravi B *et al.* Benchmarking solutions to the T-cell receptor epitope prediction problem: IMMREP22 workshop report. *ImmunoInformatics* 2023;**9**:100024.

Montemurro A, Schuster V, Povlsen HR *et al.* NetTCR-2.0 enables accurate prediction of TCR-peptide binding by using paired TCRα and *β* sequence data. *Commun Biol* 2021;**4**:1–13.

Moris P, De Pauw J, Postovskaya A *et al.* Current challenges for unseen-epitope TCR interaction prediction and a new perspective derived from image classification. *Brief Bioinform* 2021;**22**:bbaa318.

Peng X, Lei Y, Feng P *et al.* Characterizing the interaction conformation between t-cell receptors and epitopes with deep learning. *Nat Mach Intell* 2023;**5**:395–407.

Pham M-DN, Nguyen T-N, Tran LS *et al.* epiTCR: a highly sensitive predictor for TCR–peptide binding. *Bioinformatics* 2023;**39**: btad284.

Rock KL, Reits E, Neefjes J *et al.* Present yourself! by MHC class I and MHC class II molecules. *Trends Immunol* 2016;**37**:724–37.

Rudolph MG, Wilson IA. The specificity of TCR/pMHC interaction. *Curr Opin Immunol* 2002;**14**:52–65.

Scaviner D, Lefranc M-P. The human T cell receptor alpha joining (TRAJ) genes. *Exp Clin Immunogenet* 2000a;**17**:97–106.

Scaviner D, Lefranc M-P. The human T cell receptor alpha variable (TRAV) genes. *Exp Clin Immunogenet* 2000b;**17**:83–96.

Sidhom J-W, Larman HB, Pardoll DM *et al.* DeepTCR is a deep learning framework for revealing sequence concepts within T-cell repertoires. *Nat Commun* 2021;**12**:1605–12.

Springer I, Besser H, Tickotsky-Moskovitz N *et al.* Prediction of specific TCR-peptide binding from large dictionaries of TCR-peptide pairs. *Front Immunol* 2020;**11**:1803. page

Springer I, Tickotsky N, Louzoun Y *et al.* Contribution of T cell receptor alpha and beta CDR3, MHC typing, V and J genes to peptide binding prediction. *Front Immunol* 2021;**12**:664514.

Tong Y, Wang J, Zheng T *et al.* SETE: sequence-based ensemble learning approach for TCR epitope binding prediction. *Comput Biol Chem* 2020;**87**:107281.

Valkiers S, de Vrij N, Gielis S *et al.* Recent advances in T-cell receptor repertoire analysis: bridging the gap with multimodal single-cell RNA sequencing. *ImmunoInformatics* 2022;**5**:100009.

Vig J, Madani A, Varshney LR *et al.* {BERT}ology meets biology: interpreting attention in protein language models. In: *International Conference on Learning Representations, Virtual*. 2021.

Weber A, Born J, Rodriguez Martínez M *et al.* TITAN: T-cell receptor specificity prediction with bimodal attention networks. *Bioinformatics* 2021;**37**:i237–44.

Wooldridge L, Ekeruche-Makinde J, van den Berg HA *et al.* A single autoimmune T cell receptor recognizes more than a million different peptides. *J Biol Chem* 2012;**287**:1168–77.

Wu K, Yost KE, Daniel B *et al.* TCR-bert: learning the grammar of T-cell receptors for flexible antigen-xbinding analyses. bioRxiv, 2021. https://doi.org/10.1101/2021.11.18.469186.

Yang X, Garner LI, Zvyagin IV *et al.* Autoimmunity-associated T cell receptors recognize HLA-B 27-bound peptides. *Nature* 2022;**612**: 771–7.

Zhang W, Hawkins PG, He J *et al.* A framework for highly multiplexed dextramer mapping and prediction of T cell receptor sequences to antigen specificity. *Sci Adv* 2021;**7**:eabf5835.