OXFORD

Briefings in Bioinformatics, 2023, **24(2)**, 1–10
https://doi.org/10.1093/bib/bbad086
Advance access publication date 11 March 2023
Problem Solving Protocol

TEINet: a deep learning framework for prediction of TCR–epitope binding specificity

Yuepeng Jiang, Miaozhe Huo and Shuai Cheng Li

Corresponding author: Shuai Cheng Li, Department of Computer Science, City University of Hong Kong. Tel.: +852-3442-9412; Fax: +852-3442-0503; shuaicli@cityu.edu.hk

Abstract

The adaptive immune response to foreign antigens is initiated by T-cell receptor (TCR) recognition on the antigens. Recent experimental advances have enabled the generation of a large amount of TCR data and their cognate antigenic targets, allowing machine learning models to predict the binding specificity of TCRs. In this work, we present TEINet, a deep learning framework that utilizes transfer learning to address this prediction problem. TEINet employs two separately pretrained encoders to transform TCR and epitope sequences into numerical vectors, which are subsequently fed into a fully connected neural network to predict their binding specificities. A major challenge for binding specificity prediction is the lack of a unified approach to sampling negative data. Here, we first assess the current negative sampling approaches comprehensively and suggest that the *Unified Epitope* is the most suitable one. Subsequently, we compare TEINet with three baseline methods and observe that TEINet achieves an average AUROC of 0.760, which outperforms baseline methods by 6.4–26%. Furthermore, we investigate the impacts of the pretraining step and notice that excessive pretraining may lower its transferability to the final prediction task. Our results and analysis show that TEINet can make an accurate prediction using only the TCR sequence (CDR3β) and the epitope sequence, providing novel insights to understand the interactions between TCRs and epitopes.

Keywords: T cell receptor, epitope specificity, immunoinformatics, deep learning, transfer learning

INTRODUCTION

T cells are critical for the adaptive immune system, providing protection against a wide range of pathogens. To recruit T cells in an immune response, the T cell receptors (TCRs) on their surface have to recognize a non-self-immunogenic peptide (epitope) presented in the context of major histocompatibility complex molecules (MHC). The generation of these protein receptors arises mainly from the quasirandom somatic V(D)J recombination process which theoretically can produce extremely high TCR diversity of 10^{15}–10^{20} in an individual, each with unique recognition capacity for antigens [1]. Understanding the mechanisms that govern the interaction between TCR and peptide-MHC (pMHC) is considered an essential step toward personalized immunotherapy and the development of targeted vaccines.

Recent advancements in the high-throughput tetramer-associated TCR sequencing technique [2] and other experimental approaches such as tetramer analysis [3] and T-scan [4] have enabled the generation of an increasing amount of data recording the bindings of TCRs and epitopes. More and more interaction pairs are consistently being generated and stored in publicly available databases such as VDJdb [5], IEDB [6] and McPAS-TCR [7]. However, the available data are still scant compared with the theoretical TCR diversity. Further, the TCR–epitope paired data are imbalanced, as a single epitope is often linked by many TCRs. Both of them pose challenges to the development of *in silico* predictive methods.

Machine-learning-based methods are able to capture the potential laws of TCR–epitope binding from a large amount of experimental data. With the help of advanced machine learning models, several computational methods have been proposed to assess the binding of a TCR and a pMHC (epitope). Previously, a branch of research focused on designing epitope-specific models with the aim of learning the pattern of TCRs binding to the same epitope. These models range from simple sequence-alignment-based methods [8] to more complex machine learning models including random forest (e.g. TCRex [9]) and the Gaussian process classifier TCRGP [10]. However, they all share two downsides: each epitope needs a specific model trained separately; each model requires abundant training samples of epitope-specific TCRs, which are not always readily available.

To fulfill the need to predict the binding specificity of any TCR–epitope pair, previous studies have proposed generic models, which aim to predict the interaction between any pair of TCRs (CDR3β) and the pMHCs (epitope) [11–17]. These generic models can fully capitalize on the currently available paired data to unlock the binding patterns between TCRs and epitopes, and transfer the knowledge learned from paired samples of epitopes with sufficient binding TCRs to those with sparse linking TCRs. Current models have shown moderate predictive performance and demonstrated promising potential in understanding cancer progression, prognosis and responsiveness to immunotherapy. For example, Moris *et al.* [11] proposed a convolutional neural network

Yuepeng Jiang is a Ph.D. candidate in the Department of Computer Science at City University of Hong Kong. He studies machine learning and immunoinformatics.
Miaozhe Huo is a Ph.D. candidate in the Department of Computer Science at City University of Hong Kong. She studies machine learning and bioinformatics.
Shuai Cheng Li is an associate professor in the Department of Computer Science at City University of Hong Kong. His research areas include algorithms, machine learning and omics data analysis.
Received: October 21, 2022. **Revised:** February 10, 2023. **Accepted:** February 14, 2023

(CNN)-based model ImRex; Weber *et al.* introduced TITAN that encodes epitopes at the atomic level with SMILES sequences using a pretrained deep learning model. Moreover, Lu *et al.* presented pMTnet [14] that encodes TCRs and pMHCs by two respective pretrained deep learning models and applied pMTnet to investigate tumor progression and response to immunotherapy treatment. In particular, transfer learning is becoming a prior technique to develop advanced deep learning models for binding prediction since it helps leverage the knowledge from other pretraining tasks with abundant data. For instance, TITAN, NetTCR [15] and pMTnet all utilize pretrained encoders. However, the impact of the pretraining step on the final performance of predicting TCR specificity remains undiscovered.

To train and evaluate supervised models, both positive and negative samples (TCRs and epitopes that do not interact with each other) are required. However, the public TCR–epitope interaction datasets only collect positive samples, which potentially poses a challenge in model training and evaluation. The method of generating negative samples based on the existing TCR and epitope pairs directly affects the model performance. Currently, there are four major strategies for generating negative samples: (1) *Reference TCR* [15, 18]; (2) *Random TCR* [14]; (3) *Random Epitope* [12, 16, 19]; (4) *Unified Epitope* [11, 20]. Different machine learning models might adopt different negative sampling strategies, which make it difficult to fairly compare their performance. More importantly, which strategy leads to a better generalized model has not been explored and remains an open question.

In this work, we present TEINet for the prediction of the specificity of TCR binding, using the CDR3β chain of TCR and the epitope sequence within the pMHC complex. Following the concept of transfer learning, TEINet employs two separate pretrained encoders to convert TCRs and epitopes into numerical vectors, utilizing the architecture of recurrent neural networks to handle a variety of sequence lengths. We first contrast the four negative sampling strategies applied in the previous work to select the superior one. Next, we systematically validated TEINet using a large-scale TCR–epitope paired dataset and two independent validation datasets. The results demonstrated the enhancement in accuracy made over previous work. We also investigated the impact of the pretraining step on the final binding specificity prediction task. Overall, TEINet serves as a reliable computational tool for addressing the long-standing problem of predicting the TCR–epitope interaction.

METHODS

Dataset

The CDR3 regions of TCRβ chains are located in the center of the paratope and are considered as the key determinant of specificity in antigen recognition [21]. Although CDR3-α and -β synergistically drive TCR–epitope recognition [17, 22, 23], the currently available databases still record mostly βchain paired samples. Thus, we restrict ourselves to CDR3β chain sequences in this study. Besides, with the aim of developing a general model that is suitable for most cases, we took the epitope sequence inside the pMHC complex as its representation. In order to construct a large and diverse dataset, we combined the data recorded in VDJdb database [5], McPAS database [7] and the data collected by Lu *et al.* [14] together.

The data from VDJdb were downloaded from its public website (https://vdjdb.cdr3.net/) on 5 April 2022. It consists of 89 321 curated pairs of CDR3 α/β sequences along with their binding epitopes and MHC classes, covering three species. We selected only human TCR sequences, removed duplicate cases, restricted only MHC class I entries and only kept the CDR3β and epitope sequences whose lengths lie between 5–30 and 7–15 amino acids, respectively. After all these filterings, this dataset was reduced to 35 560 unique CDR3β-epitope pairs, among which 33 258 TCRs are assigned to 159 epitopes.

The McPAS-TCR dataset [7] originally contains 39 664 pairs (http://friedmanlab.weizmann.ac.il/McPAS-TCR/) and Lu *et al.* collected a total of 32 607 paring data from a series of previous publications and four chromium single-cell immune profiling solution datasets. We performed the same preprocessing step on these two datasets and removed all TCR sequences with ambiguous amino acids (B, J, O, U, X). Then, these three datasets were merged together, followed by two additional filtering steps: removal of duplicate pairs and exclusion of epitopes with less than 10 associated TCR sequences since this merged dataset is highly imbalanced. At last, we constructed a large dataset with 44 682 pairs of TCRs and epitopes, among which 41 610 TCRs are linked to 180 epitopes. An overview of the dataset is shown in Figure 1.

Negative sampling strategies

Since the TCR–epitope dataset contains only positive samples, in order to train a generalized and robust supervised model, the negative samples are required and should be generated via a biologically and computationally plausible manner to serve as an unbiased estimate of the actual distribution of non-binding pairs. For a positive sample $d_i = (e_i, t_i) \in D = \{d_i\}_{i=1}^{N}$, where e_i and t_i are the interacting epitope and TCR for sample i, the corresponding negative samples can be generated through four major sampling strategies (Figure 2A):

- *Reference TCR*. In this setting, each epitope e_i is combined with TCRs that are sampled uniformly from a reference TCR dataset $R = \{t_j\}$. The negative samples for e_i are then represented as $n_i = \{(e_i, t_j)\}_{j=1}^{M}$, where $t_j \in R$ and M is the number of negatives samples for a given positive sample [15, 18]. This approach stands upon the assumption that TCRs from the reference dataset are unlikely to bind epitopes in the positive dataset. The reference TCRs were obtained from Montemurro *et al.* [15] where these TCRs had been exposed to all tested pMHC multimers and no binding signals were detected.
- *Random TCR*. For this sampling approach, the negative TCRs for e_i are sampled uniformly from the set of TCRs $T = \{t_i\}$ in the positive binding pairs while excluding its known true TCR binding partner(s) [14]. The negative samples for e_i are then represented as $n_i = \{(e_i, t_k)\}_{k=1}^{M}$, where $t_k \in T$ and $(e_i, t_k) \notin D$.
- *Random Epitope*. In this strategy, each TCR t_i is combined with epitopes sampled uniformly from all epitopes $E = \{e_i\}$ without its true epitope binder(s) [12, 16, 19]. The sampled negative pairs for t_i are $n_i = \{(e_j, t_i)\}_{j=1}^{M}$ with $e_j \in E$ and $(e_j, t_i) \notin D$.
- *Unified Epitope*. Compared with *Random Epitope*, the only difference of *Unified Epitope* is that the epitopes are sampled according to their frequency distributions in the positive dataset [11, 20]; i.e. $n_i = \{(e_j, t_i)\}_{j=1}^{M}$ and $P_{pos}(e_j) \approx P_{neg}(e_j)$. This strategy ensures that the frequencies of epitopes are unified in the negative data and positive data.

A systematical comparison between these four negative sampling strategies is an urgent need for benchmarking different models and guiding the development of accurate and generalized models in future works. To address this demand, we resorted to the field of recommender system (RS) since the implicit feedback datasets such as the purchase history and browsing history in RS

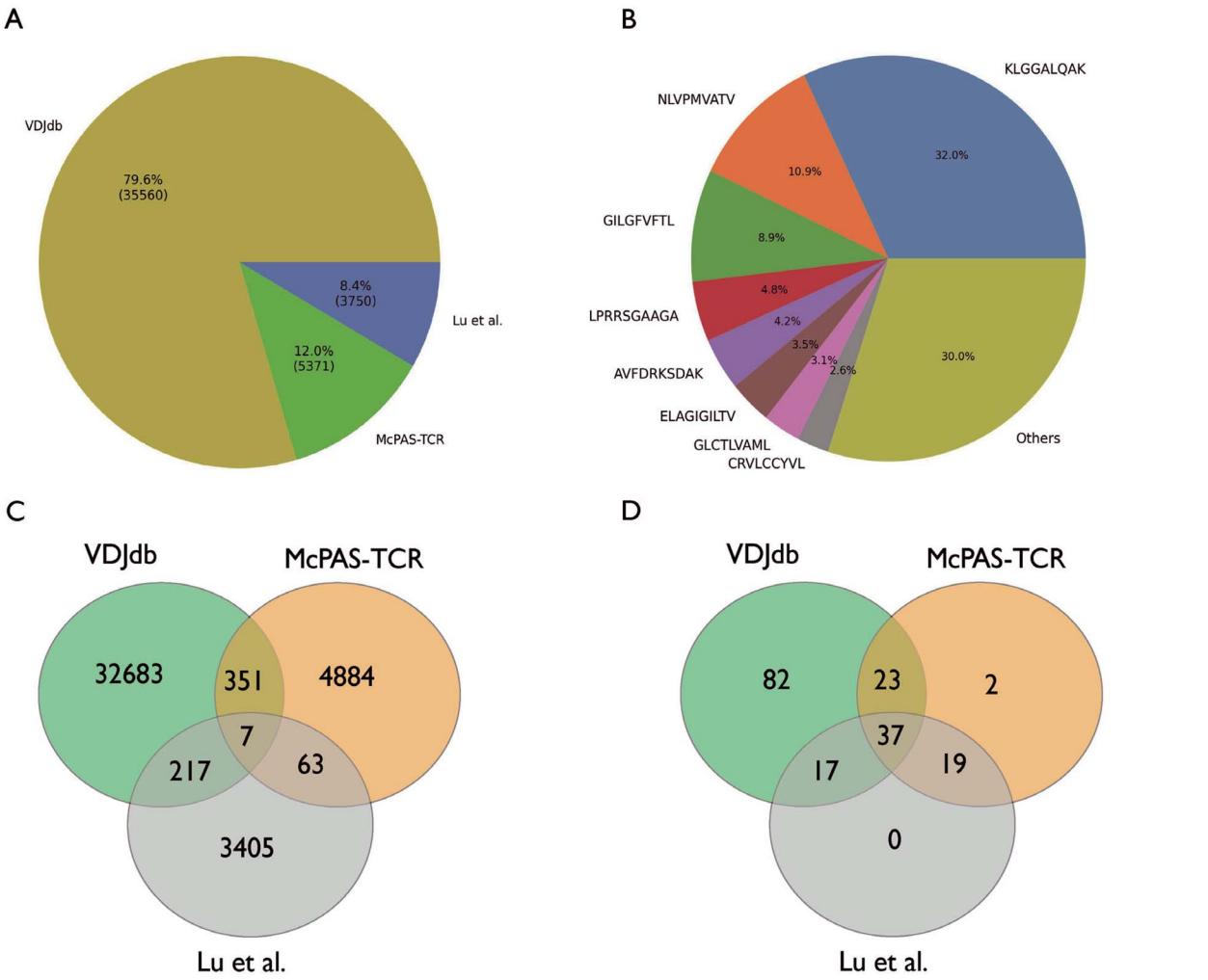

Figure 1. Overview of our constructed dataset. (**A**) The source of the paired samples in the dataset. (**B**) The number of the epitope-associated TCRs. Most TCRs are linked to a small group of epitopes. (**C**) and (**D**) Venn diagrams showing the number of (**C**) TCRs and (**D**) epitopes contributed by each source dataset.

also only record the positive interactions. Thus, we selected three evaluation metrics that are commonly used for evaluating the implicit recommendation [24, 25] and can be calculated without the attendance of negative samples.

Precision@k and Recall@k. These two metrics measure the exactness and completeness of the top k binding predictions for a given TCR. Assume that a TCR t_i in the test set $\{(e_i, t_i)\}_{i=1}^{N}$ can bind to a number of b_i epitopes (due to cross-reactivity), and a number of m_i true interacting pairs $\{(e_j, t_j)\}_{j=1}^{m_i}$ lie in the top k predictions, then these two metrics are defined as follows:

$$Precision@k = \frac{1}{N}\sum_{i=1}^{N}\frac{m_i}{k} \tag{1}$$

$$Recall@k = \frac{1}{N}\sum_{i=1}^{N}\frac{m_i}{b_i}, \tag{2}$$

where N is the total number of TCRs in the test set. A higher value of *Precision@k* indicates that more true binding pairs can be found among the top k predicted pairs; a higher value of *Recall@k* suggests a higher proportion of predicted binding pairs over all the true binding pairs.

NDCG@k. The previous two metrics overlook the order of the predictions since the ranking of the true predicted binding pairs does not affect the values of both metrics as long as they are in the top k predictions. The Normalized Discounted Cumulative Gain (NDCG) measures how relevant the predictions are and how good the ordering is, which is calculated by

$$NDCG@k = \frac{DCG@k}{IDCG@k}, \tag{3}$$

where the definitions and formulas of DCG@k and IDCG@k are described in Supplementary Text S1. Overall, these three metrics are complementary to each other and help to determine the superiority of the four negative sampling strategies. A higher value of any of the three metrics indicates a better model performance.

Pretrained encoder

To numerically encode TCR and epitopes, we capitalized on the transfer learning technique. We have earlier proposed an autoencoder model TCRpeg [26] that utilizes a recurrent neural network with GRU layers to characterize the TCR repertoires and demonstrated that it can produce high-quality vector encodings for TCR sequences. As an autoencoder model, TCRpeg is capable of capturing key features of sequence input via unsupervised

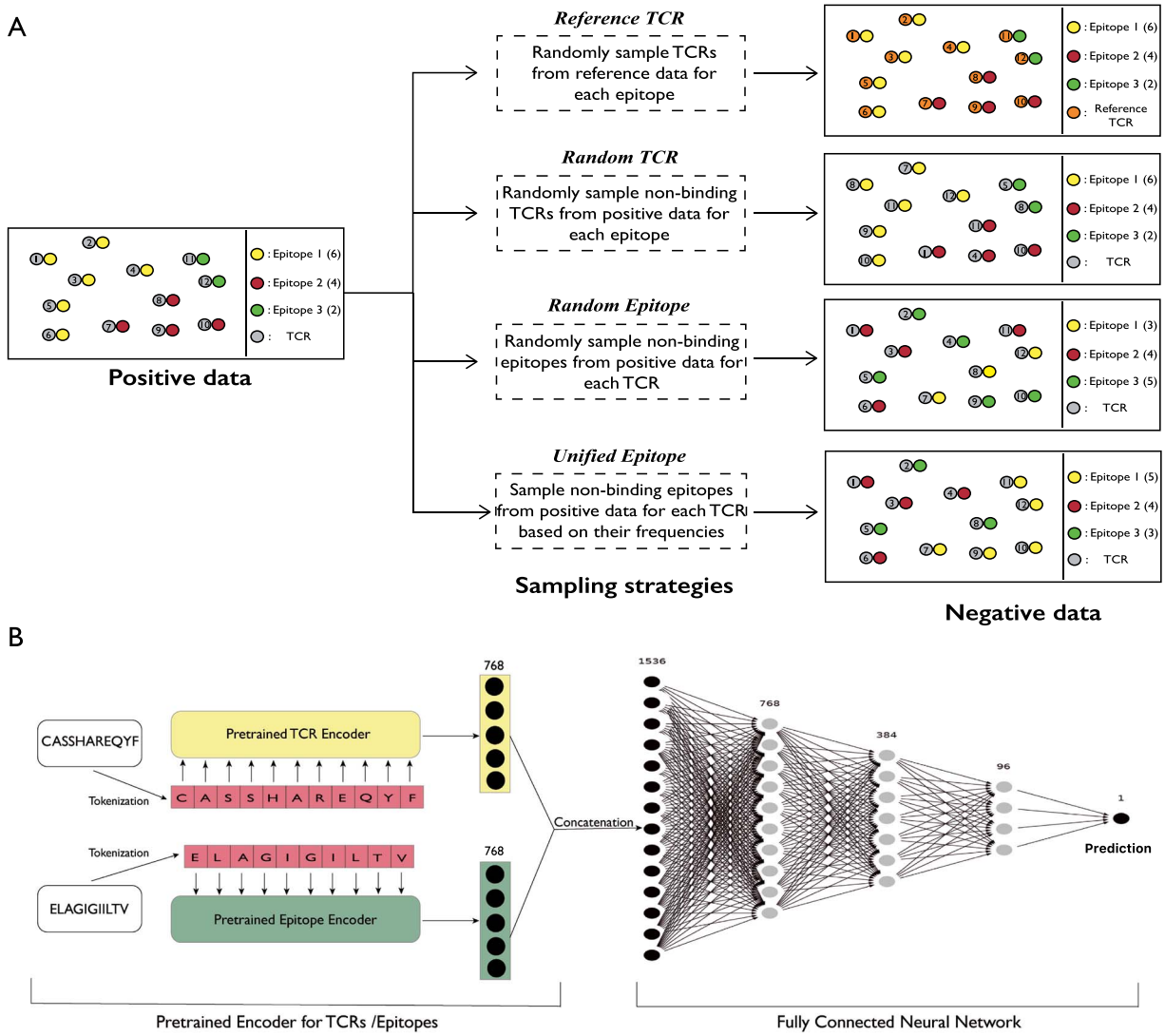

Figure 2. Illustration of each negative sampling strategy and the overall workflow of TEINet. (**A**) Sketch map of the four negative sampling strategies. In this example, there are in total 12 TCR–epitope binding pairs, with three different epitopes (depicted in yellow, red and green) linking to six, four and two TCRs, respectively. For *Reference TCR* strategy, the TCRs are randomly sampled from a reference TCR dataset inside which TCRs are considered unable to bind epitopes in the positive data. Here, we choose to generate the same number of pairs in negative data for demonstration. (**B**) General workflow of TEINet. TEINet is a two-stage deep learning model using transfer learning. At the first pretraining stage, two TCRpeg models are trained separately to learn the sequence pattern of TCRs and epitopes, and produce numerical encodings for them when the pretraining process is completed. At the next stage, encodings of TCRs and epitopes are concatenated together and output into an FCN to leverage the information from each part and make predictions accordingly.

learning of mapping between the latent space and sequence space, and more importantly, using TCRpeg for pretraining only needs plain amino acid sequences which are currently abundant. In addition, unlike the encoders in TITAN or pMTnet, TCRpeg can process sequences of arbitrary lengths without the need to pad them to a fixed length. Thus, we decided to employ two separate pretrained TCRpeg models as the encoders for TCRs and epitopes, respectively. A detailed description of TCRpeg is given in Supplementary Text S2.

To pretrain TCRpeg for encoding TCRs (TCRpeg-TCR), we fed TCRpeg with 10^6 TCR sequences collected from Emerson *et al.* [27]. We set the feature size of TCRpeg to 768 and trained it for 20 epochs by minimizing the cross-entropy loss between the output soft-maxed logits and the one-hot encoded representation of the input sequences. For encodings of epitopes, we trained another TCRpeg model (TCRpeg-Epi) with the identical architecture of TCRpeg-TCR using 362 456 unique epitope sequences collected from Mei *et al.* [28] with lengths ranging from 8 to 14 amino acids. Details on the pretraining process of TCRpeg are elaborated in Supplementary Text S3.

Model architecture

Figure 2B delineates an overview of the architecture of TEINet. Conceptually, the complex task of predicting the TCR–epitope interaction is decomposed into two steps to lower the difficulty level of the final prediction task. First, two encoding networks are pretrained so that the amino acid sequences of TCRs and epitopes can be represented by numerical vectors. Next, we concatenated these two vector encodings to form the final representations for TCR–epitope pairs. In the final step, we built a fully connected neural network (FCN) on top of these combined vector encodings to fuse the knowledge extracted from TCRs and epitopes. Specifically, the FCN consists of three hidden layers with 768, 384 and 96 neurons with the dropout [29] rate set to 0.15

Table 1. The Precision, Recall and NDCG of each negative sampling method. Performance is shown with the best performance among *Reference TCR*, *Random TCR* and *Unified Epitope* in bold

Method	Precision@3	Recall@3	NDCG@3	Precision@10	Recall@10	NDCG@10
Reference TCR	0.093±0.002	0.275±0.007	0.255±0.006	0.036±0.001	0.356±0.009	0.284±0.007
Random TCR	0.085±0.002	0.251±0.004	0.226±0.004	0.033±0.001	0.322±0.003	0.252±0.003
Unified Epitope	**0.129±0.004**	**0.380±0.012**	**0.334±0.011**	**0.052±0.001**	**0.506±0.014**	**0.380±0.012**
Random Epitope	0.192±0.002	0.567±0.006	0.484±0.006	0.081±0.001	0.788±0.002	0.565±0.005

(Supplementary Figure S4). Before feature concatenation, we employed layer normalization [30] to numerically stabilize each group of features. All neurons use the scaled exponential linear unit (SELU [31]) activation function, except for the output neuron which applies the sigmoid activation function.

Model training

TEINet was implemented in Python 3.6 and built on the deep learning framework PyTorch [32]. TEINet was trained and evaluated by a 5-fold cross-validation (CV) procedure. Note that we first split the positive data into training and validation data sets for each CV fold and then independently sampled their respective negative samples using the aforementioned strategies to avoid potential issue of data leakage. Instead of inferring TEINet on a static dataset with negative pairs sampled prior to the training process, we adopted a dynamic sampling strategy: the negative examples are sampled on the fly based on positive pairs of the training set at each training step using the sampling strategies described in the previous section. This dynamic sampling strategy demonstrates improved performance over static training (Supplementary Figure S1). For all experiments in this work, the negative pairs were sampled 10 times more than the positive pairs. TEINet optimized binary cross entropy loss with Adam algorithm [33] and an initial learning rate of 1×10^{-3}. The model was trained for 30 epochs with a batch size of 48. The learning rate was reduced at the 21st and 27th epoch by a factor of 0.1.

RESULTS

Comparison of different negative sampling strategies

We trained TEINet with each negative sampling method and observed that they could obtain performance in different scales (Supplementary Figure S2). For example, using *Reference TCR* leads to an average AUROC (area under the receiver operating characteristic) of 0.797, whereas the performance achieves an AUROC of 0.934 under *Random Epitope*, which is unexpectedly high yet useless.

We first compared the three negative sampling methods: *Random TCR*, *Reference TCR* and *Unified Epitope*. The negative data generated by these methods possess similar frequency distributions of epitopes with those in the positive data. Table 1 shows the Precision, Recall and NDCG of each schema using the TEINet. These results first demonstrated that *Random TCR* and *Reference TCR* obtained similar performance, indicating that sampling TCRs from the reference TCR pool or TCRs in positive data have a comparable effect on the model training. *Reference TCR* is slightly better than *Random TCR*, as TCRs drawn from another sequence pool constructed from healthy donors are less likely to interact with epitopes than shuffled TCRs from the positive data; i.e. *Random TCR* might produce more false negative pairs. *Unified Epitope* achieved superior performance among these three strategies by a large margin. It indicates that *Unified Epitope* can help develop a more robust and generalized model for the TCR–epitope interaction prediction task.

We next contrasted *Unified Epitope* with *Random Epitope*. It seems that *Random Epitope* is a perfect sampling strategy since it achieved extremely high values of Precision, Recall and NDCG (Table 1), and achieved an average AUROC of 0.934. However, these high values are overestimated and misleading due to the inherent imbalance of the data. Note that the number of epitope-interacting TCRs follows an extremly long-tail distribution (Figure 3A, 1B and Supplementary Figure S3) that most TCRs (70%) are associated with the top 5% epitopes. As a result, *Random Epitope* would produce skewed negative data that the majority of epitopes were matched with far more negative TCRs than positive TCRs (Supplementary Figure S3). Trained with such a skewed dataset, TEINet was driven to make predictions based on the epitope sequences without the participation of TCRs, as discussed in Dens *et al.* [34]. That is, when the input pairs consist of frequent epitopes, the model tends to predict '1s', and conversely, it is likely to predict '0s' when encountering pairs with infrequent epitopes. Thus, TEINet with *Random Epitope* obtains a misleading high performance due to the following two reasons: (1) for TCRs, TEINet often predicts high scores when they are linked to frequent epitopes, which results in high Precision, Recall and NDCG since frequent epitopes appear in most paired samples; (2) for epitopes, TEINet tends to predict high scores for pairs with frequent epitopes that possess abundant positive binding TCRs and sparse negative binding TCRs, and low scores for pairs with rare epitopes that are linked to abundant negative TCRs and sparse positive binding TCRs, which leads to high AUROC. Indeed, TEINet with *Random Epitope* obtained high prediction scores for both positive and negative pairs with frequent epitopes (Figure 3B). For instance, it outputs an average prediction score of 0.99 and 0.93 for respective positive and negative pairs of the most frequent epitope (KLGGALQAK). As a result, those negative pairs will be classified as false positives in the generic performance evaluation. Moreover, due to the long-tail distribution of the epitope-associated TCRs, *Random Epitope* will generate far fewer negative pairs than positive pairs. Thus, those false positives have a minor impact on the generic performance evaluation, resulting in a misleadingly high AUROC.

Overall, our results and analysis demonstrate that *Unified Epitope* is more appropriate for negative sampling in the TCR–epitope prediction task, which is further supported in the evaluation of independent datasets (see the following section). To eliminate potential model bias introduced by TEINet, we performed the same experiments using the ImRex model and obtained similar results (Supplementary Table S1). In the remaining experiments, *Unified Epitope* is selected as the default strategy.

Performance of TEINet

To assess the performance of TEINet, we compared it with three existing approaches: ImRex [11], TITAN [12] and pMTnet [14].

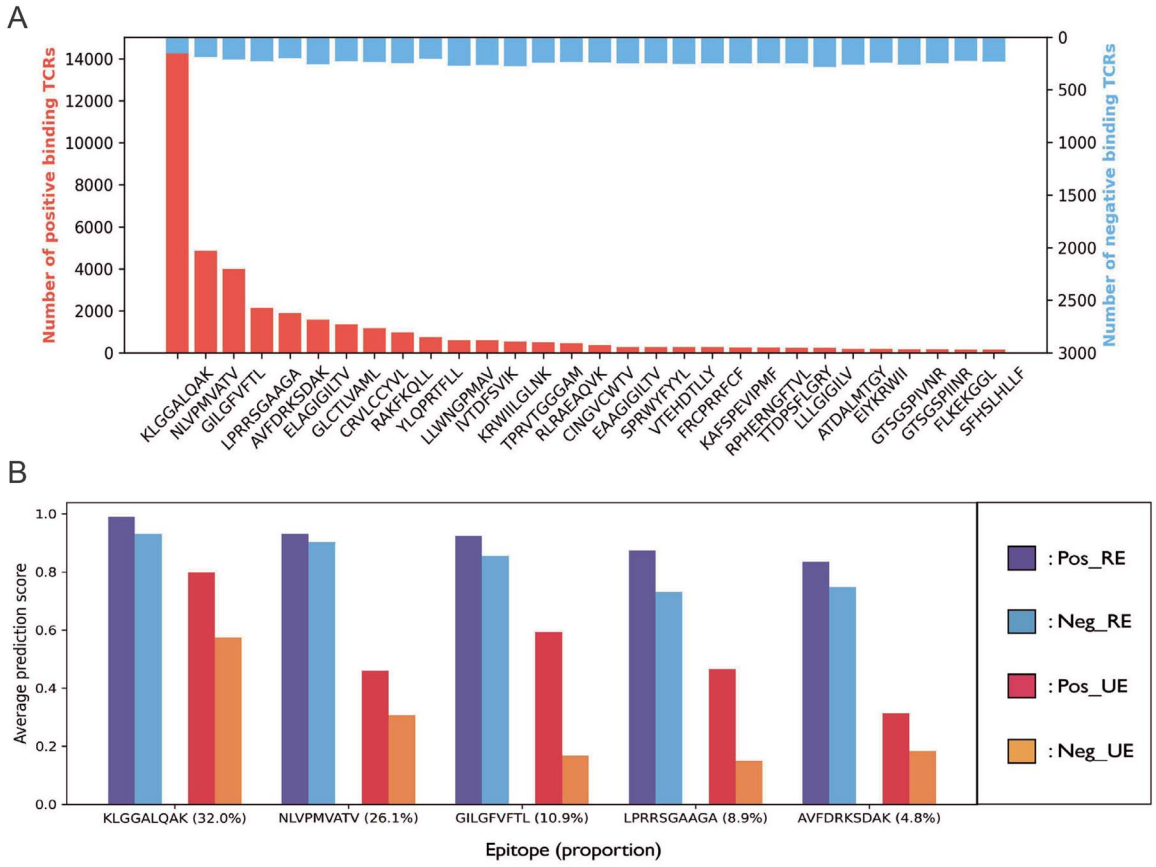

Figure 3. Distribution of the number of epitope-associated TCRs and the average prediction scores for them. (**A**) Distribution of the number of positive and negative TCRs sampled by *Random Epitope* for the 30 most abundant epitopes. Given that the epitopes are sampled randomly for each TCR and the epitope-associated TCRs follow an extreme long-tail distribution, there are far fewer negative samples than positive samples for abundant epitopes, whereas for most epitopes, there are far more negative samples than positive samples. (**B**) The average prediction scores for the positive and negative pairs of the top five most abundant epitopes. 'Pos' and 'Neg' stand for positive and negative samples; 'RE' and 'UE' represent *Random Epitope* and *Unified Epitope*. We observed that for both positive and negative pairs of abundant epitopes, *Random Epitope* will produce high predictive scores. Such a problem is greatly relieved by *Unified Epitope*.

ImRex encodes TCRs and epitopes based on their physicochemical properties and utilizes a CNN to process the combined encodings. Similar to our proposed TEINet, TITAN and pMTnet both use the pretrained encoders. The pMTnet additionally incorporates the information of the MHC allele associated with the epitope to make the prediction. More details of these models can be found in Supplementary Text S4.

Figure 4A and 4B show the AUROC and AUPRC of TEINet as well as the three baseline models. TEINet outperforms the baseline methods with an AUROC of 0.760 and an AUPRC of 0.321, while the second-best comparative model ImRex has an AUROC of 0.714 and an AUPRC of 0.269. Moreover, we calculated the Precision, Recall and NDCG of ImRex and still observed superior performance of TEINet (Supplementary Table S1). With learnable encoders that possess the capability of processing sequences in any length, TEINet can better extract sequence information and consequently make more accurate predictions. To investigate whether the superiority of TEINet remains when the similarity of TCRs between training and evaluation datasets decreases, we filtered out pairs in the test set with specific TCRs according to the Levenstein similarity thresholds (Supplementary Text S5). Figure 4C demonstrates the corresponding performance of each model under different similarity thresholds. Again, TEINet outperforms other baseline models. Further, to resolve the concern that pairs consisting of frequent epitopes would dominate the effects on performance, we report the per-epitope AUROC derived by evaluating paired data for one specific epitope (Figure 4D). We found no explicit correlation between the AUROC and the number of training samples, indicating that the complexities of the binding pattern for each epitope are different. Similar results were also found in Moris *et al.* [11]. Besides, TEINet is still superior, with the ImRex lagging behind for most epitopes.

Impact of pretraining

Transfer learning is becoming an integral part of the design of deep learning models for the prediction of TCR binding specificity. Recently developed models tend to employ pretrained encoders to transform amino acid sequences into vector representations [12, 14, 15, 19]. An analysis of the impact of the pretraining step is in demand to provide a better understanding of the pretrained encoders.

First, without the pretraining step, the performance of TEINet dropped significantly with an AUROC of 0.675, which demonstrated the necessity of the pretraining step. Next, we explored the influence of the TCR and epitope encoders singly and simultaneously (Figure 5A). It is clear that the pretraining of TCRs greatly enhanced the model performance, whereas the pretraining of epitopes only brought about slight and unstable improvement. Given that the diversity of TCRs (41 610 unique samples) is much higher than that of epitopes (180 unique samples), pretraining of TCRs enables them to be distributed separably in the feature space, which is more important for making a prediction. Further,

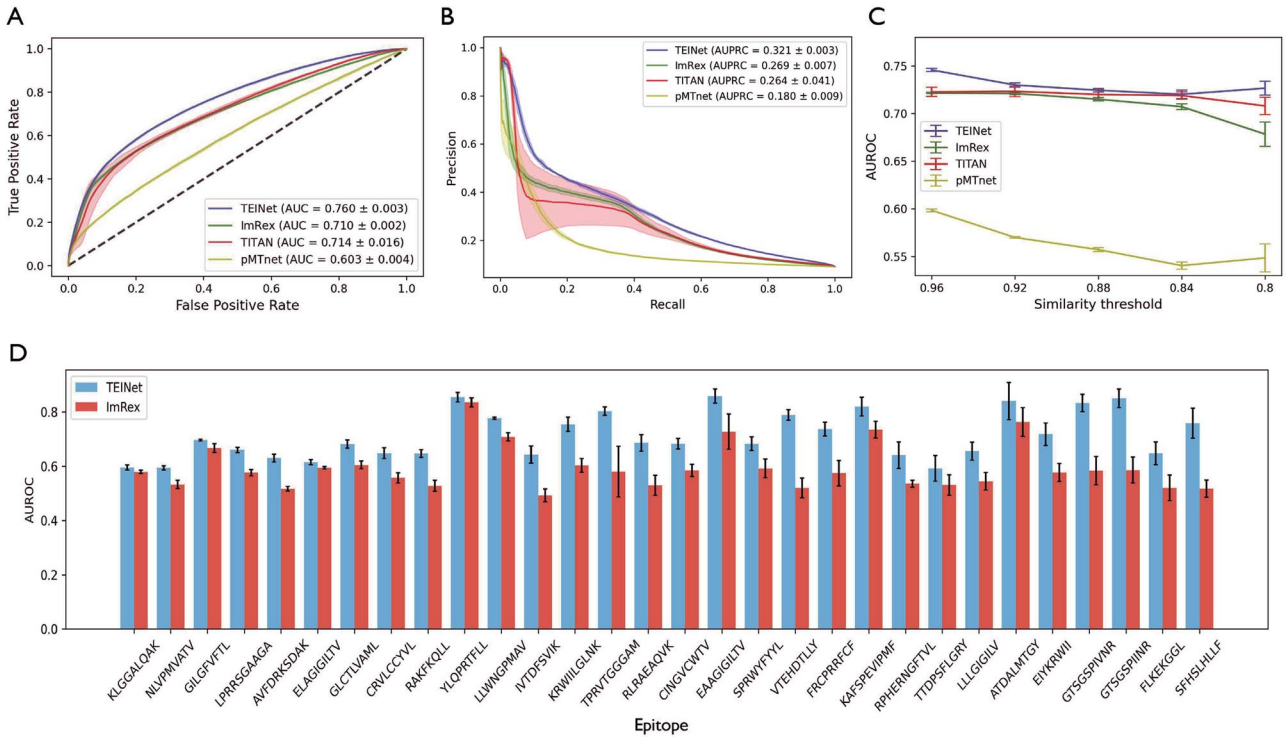

Figure 4. Performance of TEINet and the three baseline models. (**A**) The receiver operator characteristic (ROC) curves for each model. The area under the ROC curve (AUROC) values are shown in the legend. (**B**) The precision-recall (PRC) curves for each model. The area under the PRC curve (AUPRC) values are shown in the legend. (**C**) The AUROC for each model according to different similarity thresholds for filtering the test set. (**D**) The per-epitope AUROC performance for the top 30 most abundant epitopes. TEINet outperforms ImRex in these epitopes.

these two encoders improved the performance synergistically and achieved the best performance. Utilizing both pretrained encoders enhanced the AUROC by around 0.01 than using the TCR encoder alone. Notably, we observed that when the pretraining of the TCR encoder exceeded a certain epoch, the final performance slightly dropped (Figure 5A). Thus, the degree of the pretraining needs to be tuned carefully; excessive pretraining of encoders on the reconstruction task might slightly lower its transferability to the final TCR–epitope binding prediction task.

Structural analysis

Perturbation (mutational) analysis can be used to detect the important amino acid residues for the model prediction [14, 26, 35]. We grouped TCR residues by whether or not they formed any direct contact with any residue of epitopes within 5Å and assumed that substitutions inside the contact region would lead to dramatic changes in the predicted binding score. To analyze the effects of predictive models on the contact/non-contact region, we collected 105 solved TCR–epitope interacting complex structures from the public RCSB Protein Data Bank (PDB) database [36] as the ground truth data. We performed the alanine scanning technique in biophysics studies [37] on the TCRs in the PDB database using the predictive models. Figure 5B illustrates the average score difference for each model inside the contact and non-contact region. We observed that for TEINet, the contact residues were more likely to induce larger drops in predicted TCR–epitope binding strength than non-contact residues, which supports our assumption.

Evaluation on independent datasets

To further compare the predictive performance of each model, we collected two independent test sets. We selected the TBAdb [38] dataset, which includes 439 binding pairs on 414 unique TCRs and 42 epitopes as our first independent test set; The 105 interacting pairs extracted from the PDB database aforementioned were selected as the second independent test set. As before, the same filtering procedure was applied to them. Figure 5C–5F show the performance of each model on the independent test sets. Again, TEINet achieved superior performance over the other baseline methods. Note that for the PDB dataset, TEINet obtained a lower AUROC value of 0.646. We attributed it to the small overlap of epitopes as there is only one epitope in the PDB dataset that also appears in the training data. Moreover, given that the PDB dataset is an approximately balanced dataset with each epitope binding with one or two TCRs, the *Random Epitope* and *Unified Epitope* will generate similar negative data, which enables us to compare these two strategies by the AUROC value. Thus, we trained two TEINets each using *Random Epitope* or *Unified Epitope* during the training process and then evaluated them on the PDB database constructed with *Random Epitope*. We observed that TEINet trained with *Random Epitope* obtained an AUROC of 0.572, which was surpassed by *Unified Epitope* by a large margin with an AUROC of 0.644 (Supplementary Figure S5). This finding further supports the advantage of *Unified Epitope*.

DISCUSSION

The prediction of TCR specificity to epitope has been a challenging problem. The immense searching space of immune receptors, lack of curated training samples and absence of negative samples remain challenges for algorithm development. In recent years, public databases have been accumulating an enormous amount of TCR–epitope interacting data. Benefiting from the enrichment of available data, it is possible to develop accurate deep learning models to tackle the challenging task of TCR–epitope interaction prediction.

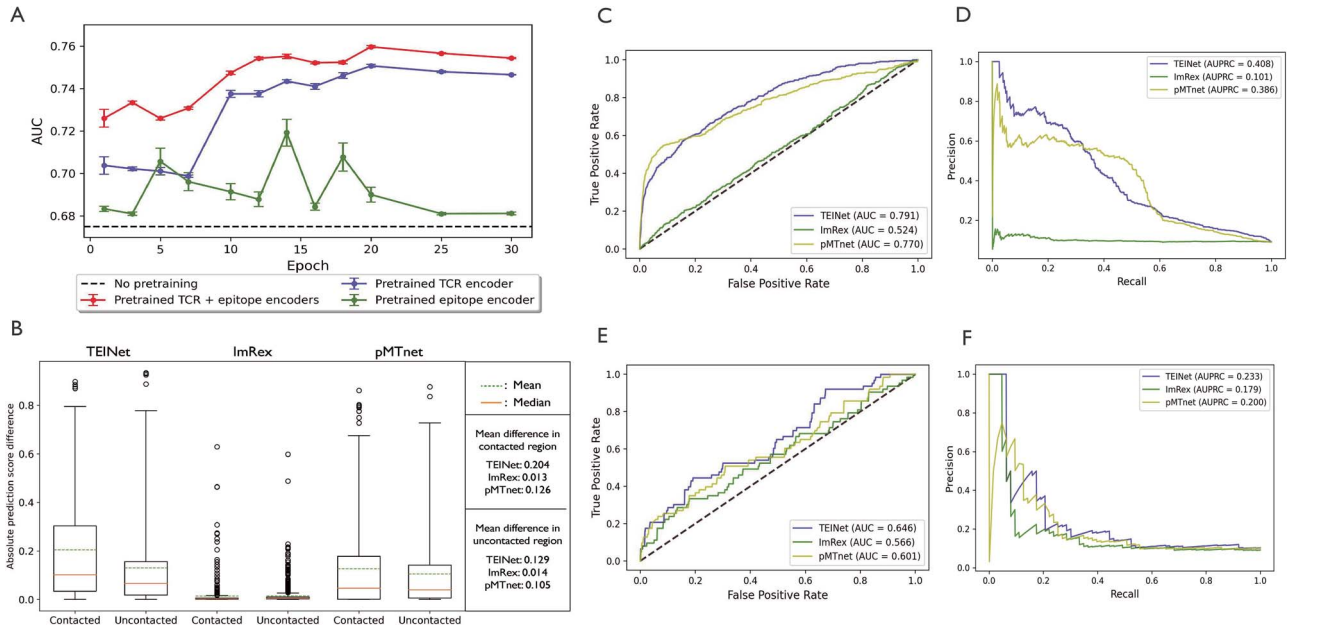

Figure 5. Investigation of the impact of the pretraining stage and further validations of TEINet. (**A**) The AUROC values with the encoders pretrained for different epochs under three different settings: only pretraining the encoder for TCRs (blue); only pretraining the encoder for epitopes (green); pretraining the encoders for both TCRs and epitopes (red). (**B**) The absolute difference in prediction scores for each model between the contacted and uncontacted residues. Using TEINet, residues with direct contacts are more likely to induce larger changes in the predicted binding strength than non-contact residues. (**C**–**F**) The ROC curves and PRC curves for models in the two independent test sets: (**C** and **D**) TBAdb and (**E** and **F**) PDB. The corresponding AUROC/AUPRC values are shown in the legend.

In this work, we have proposed TEINet, a new deep learning model for predicting the TCR binding specificity. TEINet only requires the CDR3β chain of the TCR and epitope sequence of the pMHC complex to make the prediction. Though the CDR3α chain and the MHC allele are shown to be beneficial in this task [10, 13–15, 23], the paired data are still rare compared with single-chain data, which limits the generalizability of the pair-chain model. We leave the exploration of both CDR3 chains and MHC alleles to future work. TEINet employed TCRpeg [26] to extract the sequence information of TCRs and epitopes and transform them into numerical vector space. TEINet then combined the encodings of TCRs and epitopes and used an FCN to make the final prediction, leveraging the knowledge from TCRs and epitopes.

To train and evaluate a supervised model, negative samples are required. However, currently there is no unified method for negative sampling, which poses a challenge for comparing different models. For example, *Random TCR* was applied in pMTnet [14]; *Reference TCR* was applied in NetTCR [15, 18]; *Random Epitope* was employed in TITAN [12]; *Unified Epitope* was employed in ImRex [11]. We thus proposed three metrics, Precision, Recall and NDCG, that are unrelated to negative samples to compare different sampling strategies. We manifested that *Unified Epitope* is the winner sampling method for the development of a more accurate model as discussed in previous sections. Thus, we recommend *Unified Epitope* as the default negative sampling method in future works.

To showcase the predictive strength of TEINet, we compared TEINet with another three published deep learning models: ImRex [11], TITAN [12] and pMTnet [14]. We performed the 5-fold cross-validation procedure on our constructed dataset which consists of 44 682 interacting pairs. We observed that TEINet achieved an AUROC of 0.760 and an AUPRC of 0.321 and outperformed other comparative models with the best AUROC of 0.714 and AUPRC of 0.269. Further, we also evaluated and compared these models on two additional independent test sets. Again, TEINet surpassed other baseline models.

The usage of the transfer learning technique has become a trend in the design of deep learning models for the TCR–epitope binding prediction task. Many recently published models capitalized on the pretrained encoders that leveraged the knowledge learned from other tasks with abundant data [12, 14, 15, 19]. However, the impact of the pretraining step on the final prediction accuracy remains unknown, which could potentially hinder the exploitability of pretrained encoders. Here, we disentangled the effect from each encoder (Figure 5A–5C). We first observed that the pretraining of the TCR encoder improved the TEINet by a much larger margin than that of the epitope encoder, which could be explained by the vast diversity of TCRs. More importantly, we found that excessive pretraining might reduce the transferability of the pretraining step to the final prediction task. Thus the degree of pretraining needs to be tuned carefully.

At last, we analyzed whether the prediction from TEINet can reveal the structural information of the interacting complex. We grouped residues of TCRs that form any contact with epitope within 5Å into the contact region. Contact residues should be more important than non-contact residues in forming the interaction between TCRs and epitopes [39]. Indeed, larger drops of predicted scores were observed inside the contact region than non-contact region using TEINet.

One caveat of this work is the potential bias caused by predominant epitopes and their associated TCRs in the training data. Though no explicit bias was found in Figure 4D, we expect more pairing data to be accumulated in the future. In addition, we could further inspect the methods or strategies that can help to learn a more robust model from the imbalanced data in the future, such as assigning different weights to pairing data with different epitopes. Another direction of future work lies in the exploration of the more challenging task of predicting pairs with novel

epitopes or even novel TCRs at the same time. Under this setting, which negative sampling strategy is more useful for the development of accurate models remains unclear and needs further analysis.

In summary, we have designed TEINet to predict the interaction between TCRs and their epitope targets. Our results demonstrate that TEINet achieved superior performance over three other comparative models only using the information of CDR3β chains and epitope sequences. We also compared different negative sampling strategies and suggested that *Unified Epitope* is more appropriate for the development of a generalized model. We expected that with enhanced accuracy in predicting the potential immune response of T-cells to epitopes, TEINet could be beneficial for the *in silico* design and implementation of immunotherapy in the era of personalized medicine.

Key Points

- We propose a TCR–epitope binding prediction tool TEINet that achieves better prediction accuracy than comparative baseline models.
- We summarize four methods for generating negative data and show that *Unified Epitope* is the most suitable one.
- We investigate the impact of the pretraining step and find that excessive pretraining could lower its transferability to the final prediction task.

ACKNOWLEDGMENTS

We thank all contributors to VDJdb, McPAS-TCR and other TCR specificity datasets for making their data publicly available.

FUNDING

This work was supported by strategic interdisciplinary research grant [7005215] from the City University of Hong Kong.

DATA AND CODE AVAILABILITY

TEINet was written in Python using the deep learning library Pytorch [32]. All the data and code are available at https://github.com/jiangdada1221/TEINet.

REFERENCES

1. Laydon DJ, Bangham CRM, Asquith B. Estimating t-cell repertoire diversity: limitations of classical estimators and a new approach. *Philos Trans R Soc B: Biol Sci* 2015; **370**(1675): 20140291.
2. Zhang S-Q, Ma K-Y, Schonnesen AA, *et al.*High-throughput determination of the antigen specificities of t cell receptors in single cells. *Nat Biotechnol* 2018; **36**(12): 1156–9.
3. Altman JD, Moss PAH, Goulder PJR, *et al.*Phenotypic analysis of antigen-specific t lymphocytes. *Science* 1996; **274**(5284): 94–6.
4. Kula T, Dezfulian MH, Wang CI, *et al.*T-scan: a genome-wide method for the systematic discovery of t cell epitopes. *Cell* 2019; **178**(4): 1016–28.
5. Shugay M, Bagaev DV, Zvyagin IV, *et al.*Vdjdb: a curated database of t-cell receptor sequences with known antigen specificity. *Nucleic Acids Res* 2018; **46**(D1): D419–27.
6. Vita R, Mahajan S, Overton JA, *et al.*The immune epitope database (iedb): 2018 update. *Nucleic Acids Res* 2019; **47**(D1): D339–43.
7. Tickotsky N, Sagiv T, Prilusky J, *et al.*Mcpas-tcr: a manually curated catalogue of pathology-associated t cell receptor sequences. *Bioinformatics* 2017; **33**(18): 2924–9.
8. 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.
9. 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.
10. Jokinen E, Huuhtanen J, Mustjoki S, *et al.*Predicting recognition between t cell receptors and epitopes with tcrgp. *PLoS Comput Biol* 2021; **17**(3): e1008814.
11. 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**(4): bbaa318.
12. Weber A, Born J, Martínez MR. Titan: T-cell receptor specificity prediction with bimodal attention networks. *Bioinformatics* 2021; **37**(Supplement_1): i237–44.
13. 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**(20) :eabf5835.
14. Tianshi L, Zhang Z, Zhu J, *et al.*Deep learning-based prediction of the t cell receptor–antigen binding specificity. *Nature. Mach Intell* 2021; **3**(10): 864–75.
15. 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): 1–13.
16. 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;1803.
17. Dash P, Fiore-Gartland AJ, Hertz T, *et al.*Quantifiable predictive features define epitope-specific t cell receptor repertoires. *Nature* 2017; **547**(7661): 89–93.
18. 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;433706.
19. Fang Y, Liu X, Liu H. Attention-aware contrastive learning for predicting t cell receptor-antigen binding specificity. *bioRxiv* 2022.
20. Cai M, Bang S, Lee H. Tcr-epitope binding affinity prediction using multi-head self attention model.
21. Hou X, Wang M, Chong L, *et al.*Analysis of the repertoire features of tcr beta chain cdr3 in human by high-throughput sequencing. *Cell Physiol Biochem* 2016; **39**(2): 651–67.
22. Lanzarotti E, Marcatili P, Nielsen M. T-cell receptor cognate target prediction based on paired α and β chain sequence and structural cdr loop similarities. *Front Immunol* 2019; **10**: 2080.
23. Springer I, Tickotsky N, Louzoun Y. Contribution of t cell receptor alpha and beta cdr3, mhc typing, v and j genes to peptide binding prediction. *Front Immunol* 2021; **12**:664514.
24. Bekker J, Davis J. Learning from positive and unlabeled data: a survey. *Mach Learn* 2020; **109**(4): 719–60.
25. Chen C, Ma W, Zhang M, *et al.*Revisiting negative sampling vs. non-sampling in implicit recommendation. *ACM Trans Inf Syst*.
26. Jiang Y, Li SC. Deep autoregressive generative models capture the intrinsics embedded in t-cell receptor repertoires. *bioRxiv* 2022.

27. Emerson RO, DeWitt WS, Vignali M, *et al.*Immunosequencing identifies signatures of cytomegalovirus exposure history and hla-mediated effects on the t cell repertoire. *Nat Genet* 2017; **49**(5): 659–65.
28. Mei S, Li F, Xiang D, *et al.*Anthem: a user customised tool for fast and accurate prediction of binding between peptides and hla class i molecules. *Brief Bioinform* 2021; **22**(5):bbaa415.
29. Srivastava N, Hinton G, Krizhevsky A, *et al.*Dropout: a simple way to prevent neural networks from overfitting. *J Mach Learn Res* 2014; **15**(1): 1929–58.
30. Ba JL, Kiros JR, Hinton GE. Layer normalization. *arXiv preprint arXiv:160706450* 2016.
31. Klambauer G, Unterthiner T, Mayr A, *et al.*Self-normalizing neural networks. *Adv Neural InfProcess Syst* 2017; **30**.
32. Paszke A, Gross S, Massa F, *et al.*Pytorch: an imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst* 2019; **32**.
33. Kingma DP, Jimmy BA. Adam: a method for stochastic optimization. *arXiv preprint arXiv:14126980* 2014.
34. Dens C, Bittremieux W, Affaticati F, *et al.*Interpretable deep learning to uncover the molecular binding patterns determining tcr–epitope interactions. *bioRxiv* 2022.
35. John-William Sidhom H, Larman B, Pardoll DM, *et al.*Deeptcr is a deep learning framework for revealing sequence concepts within t-cell repertoires. *Nat Commun* 2021; **12**(1): 1–12.
36. Sussman JL, Lin D, Jiang J, *et al.*Protein data bank (pdb): database of three-dimensional structural information of biological macromolecules. *Acta Crystallogr D Biol Crystallogr* 1998; **54**(6): 1078–84.
37. Weiss GA, Watanabe CK, Zhong A, Goddard A, and Sidhu SS. Rapid mapping of protein functional epitopes by combinatorial alanine scanning. *Proc Natl Acad Sci*, **97**(16): 8950–4, 2000.
38. Zhang W, Wang L, Liu K, *et al.*Pird: pan immune repertoire database. *Bioinformatics* 2020; **36**(3): 897–903.
39. Chowell D, Krishna S, Becker PD, Cocita C, Shu J, Tan X, Greenberg PD, Klavinskis LS, Blattman JN, and Anderson KS. Tcr contact residue hydrophobicity is a hallmark of immunogenic cd8+ t cell epitopes. *Proc Natl Acad Sci*, **112**(14): E1754–62, 2015.