TCR-pMHC Binding Specificity Prediction from Structure Using Graph Neural Networks

Jared K. Slone, Anja Conev, Mauricio M. Rigo, Alexandre Reuben, and Lydia E. Kavraki

***Abstract*—The mapping of T-cell-receptors (TCRs) to their cognate peptides is crucial to improving cancer immunotherapy. Numerous computational methods and machine learning tools have been proposed to aid in the task. Yet, accurately constructing this map computationally remains a difficult problem. Most prior work has sought to predict TCR-peptide-MHC (TCR-pMHC) binding specificity by analyzing the amino acid sequences of the TCRs and peptides. However, recent advancements in crystallography, cryo-EM, and *in silico* protein modeling have provided researchers with the necessary data to analyze the 3D structures of TCRs, peptides, and MHCs. Current research suggests that information contained in the 3D structure of the TCRs and pMHCs can explain instances of TCR specificity that are not explained by sequence alone. As protein structure data continues to become more accurate and easier to obtain, structure-based methodologies for predicting TCR-pMHC binding will become increasingly important. We present STAG, a novel graph-based machine learning architecture for predicting TCR-pMHC binding specificity using 3D structure data. We show that STAG achieves comparable or better performance than existing methods while utilizing only spatial and physicochemical features from modeled protein structures.**

***Index Terms*—structural bioinformatics, immunology, geometric deep learning, TCR-pMHC**

I. Introduction

The recent success of immunotherapy treatments relies largely on T-cells recognizing tumor-associated peptides presented by Major Histocompatibility Complexes (MHCs) through their T cell receptors (TCRs) [1]. Successful TCR-pMHC binding prediction can aid in immunotherapies as a first step in tumor antigen vaccine development, the design of better adoptive cell therapy, or the engineering of TCRs that target cancer more effectively. In practice, *in silico* methods for predicting TCR-pMHC interactions are often used to triage expensive and time-consuming wet-lab experiments, meaning that any increase in prediction accuracy can result in fewer failed experiments and faster development of treatments for patients. Thus, predicting TCR-peptide-MHC (TCR-pMHC) binding specificity is key to improving these cancer immunotherapy treatments. Greater understanding of what drives TCR-pMHC binding will also shed light on how the immune system functions, improving vaccine development and drug design. Accordingly, the task of predicting TCR-pMHC interactions *in silico* is referred to as the “holy grail of systems immunology” [2]. An example of a TCR-pMHC complex is shown in Figure 1.

In spite of the tremendous implications of accurately predicting TCR-pMHC binding specificity, doing so has remained a formidable task. This difficulty is largely due to the complexity of the system. There are estimated to be more than 10^7 unique TCRs in each individual [3] and TCRs may be presented with any of over 20^9 possible peptides bound to one of the over 25,000 MHCs [4]. This immense diversity leads to a dauntingly large search space when screening for potential TCR-pMHC interactions and necessitates sophisticated computational methods to aid in the task. What computational methods to use and how best to encode features from the TCR and pMHC when predicting TCR-pMHC binding specificity remains an open problem [2].

Throughout this work, we differentiate between two distinct data modalities that may be used for predicting TCR-pMHC binding patterns: 3D-protein structure data and amino acid sequence data. Like all proteins, TCRs, peptides, and MHCs may be represented by their amino acid sequences. However, there is a 3D structure to proteins once folded that has been shown to explain behaviors not captured by sequence alone [5]. Machine learning (ML) methods that use structure have achieved state-of-the-art results over sequence methods in tasks such as protein binding site prediction [6] and protein function prediction [7]. In the case of TCRs and pMHCs, it has been shown that structural attributes, such as the conformation of the peptide and the different angles at which the TCR can bind to the MHC, can account for cases of one TCR reacting with multiple pMHCs [5], [8]. This phenomenon is known as cross-reactivity and is not always apparent from sequence-based analysis [9]. Furthermore, structure-based methodologies offer greater interpretability than methods trained on sequence alone [10]. Still, current ML tools for predicting binding pairs of TCR-pMHCs are designed almost exclusively around protein sequence data. This is largely because sequence data for TCR-pMHCs can be easily obtained through advanced high throughput sequencing methodologies and is thus more abundant than structure data. Yet experimental methods for determining protein structure, such as cryo-EM, have become less expensive and more accurate in recent years. As a result, the number of solved TCR structures has nearly tripled over the last 10 years [11]. Additionally, tools such as AlphaFold2 [12] now facilitate accurate protein modeling *in silico*.

Structure-based methodologies for predicting TCR-pMHC binding will become increasingly useful as protein structure

Jared K. Slone, Anja Conev, Mauricio M. Rigo are with the Department of Computer Science, Rice University, Houston, TX, 77005, USA

Alexandre Reuben is with the Department of Thoracic/Head & Neck Medical Oncology, MD Anderson Cancer Center, Houston, TX, 77030, USA

Lydia E. Kavraki is with the Department of Computer Science, Rice University, Houston, TX, 77005 USA, and the Ken Kennedy Institute, Houston, TX, 77005 USA

Manuscript received —-; revised —-.

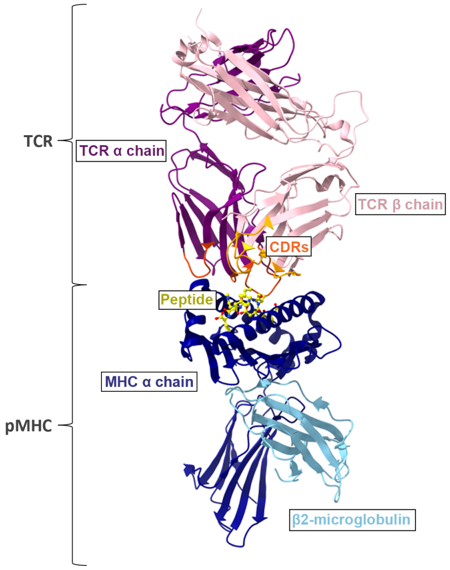

Fig. 1. Structure of the TCR-pMHC complex: TCR α chain in purple, TCR β chain in pink, CDR loops of the TCR in orange, peptide in yellow, MHC α chain in blue, and the β2-microglobulin of the MHC in cyan.

data continue to become easier to obtain and more accurate. The question is, how useful is the structure data currently available for predicting TCR-pMHC binding specificity, particularly the data that comes from *in silico* protein modeling. Our work seeks to answer this question by utilizing modeled 3D structures of TCRs and pMHCs to predict binding specificity. In order to make these predictions, we have developed a novel ML architecture that uses graph neural networks to predict TCR-pMHC binding from structure data, which we call STAG: **S**tructural **T**CR **A**nd pMHC binding specificity prediction **G**raph neural network. In this work, we evaluate STAG against other uni-modal methods for predicting TCR-pMHC binding specificity from protein structure data. We also evaluate STAG and other structure-based approaches against uni-modal methods for predicting TCR-pMHC binding specificity from amino acid sequences. We show that STAG achieves better performance in predicting TCR-peptide-MHC binding specificity than other structure-based methods and offers improved interpretability compared to existing sequence-based methods.

II. Previous Work

A. TCR-pMHC Binding Specificity Prediction

Being able to accurately predict TCR-pMHC binding *in silico* could lead to significant advancements in cancer treatment, drug design, vaccine development, and medicine as a whole. As such, there has been a large amount of work done to develop ML methods that can provide accurate and interpretable predictions for this task [13]–[35]. The overwhelming majority of these methods are uni-modal and only consider the amino acid sequence of the TCR and the peptide. Many of these methods consider only the CDR3β region of the TCR.

In this work, we compare our method to two prominent sequence-based ML models, NetTCR 2.2 [21] and ERGO II [20]. NetTCR 2.2 and ERGO II are extensions of the tools NetTCR [36] and ERGO [34] respectively, expanded to be able to train on paired TCRα chain and TCRβ chain sequences. Since information from both the α and β chains is contained in the protein structure, benchmarking against methods that only take the TCR's CDR3β chain into account would likely bias the results in favor of our method [17], [20]. Importantly, at the time of writing, ERGO II and NetTCR 2.2 are among the state-of-the-art for TCR-pMHC specificity prediction. ERGO II has been shown to have comparable or even better performance than the latest methods to predict binding specificity using large language models (LLMs) [14], depending on the datasets used.

While the majority of ML approaches to TCR-pMHC interaction prediction take amino acid sequences as input, there has been promising work that seeks to learn from the structure of the proteins as well [18], [23], [27], [29], [37]. Nearly all work taking TCR-pMHC structures as input have built their predictions around the inter-protein contacts between the TCR and the pMHC. Contacts were used alongside sequence information as part of the input to a random forest model that predicts the affinities between TCRs and pMHCs that are previously known to bind [23]. Another work trained a support vector machine on contact information and sequence data [29]. However, there is no open-source implementation of these methods that can be retrained on external data, so they will not be considered here. The tool RACER [27] uses contacts to train a custom scoring function to discriminate strong from weak binders (see Section 3 in Supplementary Material). In this work, we benchmark against the RACER protocol for predicting TCR-pMHC specificity.

B. TCR-pMHC Modeling

While there are experimentally known 3D structures for hundreds of TCR-pMHC complexes, these hardly represent enough data points to train a reliable ML model. Thankfully, our ability to accurately model TCR-pMHC complexes has increased dramatically in recent years. In this work we utilize computational models of TCRs and pMHCs as input when trying to predict TCR-pMHC binding patterns from protein structure data. Various protocols have been published that produce accurate structural models of TCR-pMHCs from sequence information [18], [38]–[40]. These make it possible to build large structure datasets, such as the ones used in this study. Owing to the high computational cost of running some of these tools, we chose to model most of the TCRs and pMHCs in this work with the higher throughput protocols TCRpMHCmodels [38] and immune-scape [39] so as to have large enough datasets to make a fair comparison when retraining the existing methods.

C. Structural Proteomics

Graph neural networks have achieved state-of-the-art performance in a variety of proteomics problems, including protein-protein docking [6], protein-ligand docking [41], protein-ligand binding affinity prediction [42], and protein function prediction [7]. All of these methods work by first encoding the protein's 3D structure as a graph where nodes represent individual atoms or amino acids in the protein. Edges between the nodes are then defined according to their distances from one another or other relationships present within the protein. Graph neural networks then utilize message passing to aggregate information between connected nodes along their edges and reason about the graph as a whole. The potential of this approach to capture the geometry and important physicochemical relationships within proteins is evident in the impressive performance of graph neural networks on various proteomics tasks, such as those mentioned above. Our model, STAG, builds on this previous work by applying a novel graph convolutional neural network to the task of TCR-pMHC binding specificity prediction.

3D convolutional neural networks (3D-CNNs) are another prominent ML architecture used with protein structure data [43]–[45]. Yet, to our knowledge there is no prior work applying 3D-CNNs to the task of TCR-pMHC binding prediction. In this work, we develop as a baseline a novel 3D-CNN pipeline that predicts TCR-pMHC binding specificity from structure data. To compare our graph-based method to 3D-CNNs, STAG was benchmarked against the 3D-CNN pipeline developed as part of this work. Further details on the implementation of the 3D-CNN pipeline can be found in Section 2 of the Supplementary Material.

D. *Establishing fair benchmarks for TCR-pMHC binding prediction with ML methods*

To date, there are no large gold standard datasets for fair benchmarking of TCR-pMHC binding predictors. Previous work that tried to benchmark different tools suffered from critical inconsistencies. These include differing biases in the training data or a limited number of unseen TCR-pMHCs that can be used to test against previously published tools. In one work, for example, a test set of only 59 TCR-pMHC pairs was reserved to compare with NetTCR [28]. As these tools are trained using tens of thousands of samples, utilizing only a few dozen data points to compare them makes it difficult to derive conclusions about their relative usefulness with a high degree of certainty. In another work, the importance of training and testing tools on the same dataset regardless of size was evident, as NetTCR achieved an AUC of only 0.518 on the testing dataset in the paper when used "out of the box" but an AUC of 0.931 when retrained using the training dataset in the paper [16]. Similar variability in performance has been reported for other tools when applied to new datasets without retraining [46]. Such drastic differences in performance make it essential to train models on identical datasets and have large mutual testing sets before drawing conclusions about their relative effectiveness in predicting TCR-pMHC binding specificity. In this work, we take care to utilize the same testing and training sets for all computational methods compared. This gives a fairer comparison of the various tools considered and allows for an accurate assessment of their relative strengths and weaknesses. We make these datasets public for others to utilize in future benchmarks.

III. Methods

A. *STAG*

STAG is a machine learning framework that utilizes a graph neural network for classifying binding and non-binding TCR-pMHC pairs from protein structure data.

1) Graph Representation: In TCR-pMHC interactions, we always know where the interaction site is (see Figures 1 & 2). The distal part of the TCR, comprised of the CDR loops, comes into contact with the peptide and the MHC cleft, as shown in Figure 1. To take full advantage of this prior knowledge, we construct an interaction graph considering only residues at the interaction site. To do this, we discard any residues from the TCR that are more than 14Å away from the nearest pMHC residue in our 3D model. We also discard any residues from the MHC that are more than 14Å away from the nearest TCR residue in our 3D model. Discarded residues are colored grey in Figure 2. Restricting our input to only include residues at the interaction site reduces the presence of noise and artifacts in the training data and improves model performance.

The TCR, peptide, and MHC are comprised of amino acids folded together in 3D space. To capture the geometry of these structures, STAG uses a graph representation $G = (V, E)$ similar to [47] where each node $v_i \in V$ represents one residue and has 3D coordinates $x_i \in \mathbb{R}^3$ centered at the carbon-α atom of the residue. Edges between nodes are defined by a radius graph of 8Å meaning that only residues less than 8Å away from one another in the 3D structure will be connected in the graph. Note that the graph is constructed only from residues at the binding site, while other residues in the complex are discarded as described above.

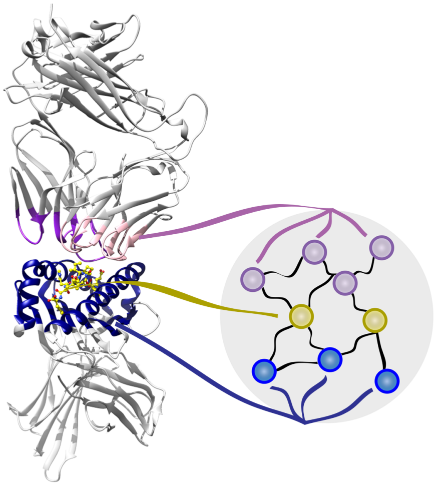

Fig. 2. Graph encoding of the interaction site for a TCR and pMHC. The structure of the TCR-pMHC complex is shown on the left. The interaction site is colored (TCR - top, pink; peptide - middle, yellow; MHC - bottom, blue). Each amino acid in the complex (left) is mapped to a node in the graph (right).

The value of each node in our graph representation of the TCR-pMHC complex is a vector encoding the physicochemical properties of the amino acid represented by that node. To encode these properties, we use the Atchly [48] and Kidera

[49] factors, vector representations of the 20 canonical amino acids derived from their physical and chemical properties. The nodes also encode parent chain information as a 1-hot vector indicating whether the node is part of the TCR-α chain, TCR-β chain, MHC, or peptide.

Edges encode the Euclidean distance between amino acids using radial basis functions. The set of radial basis functions converts scalar distance d to a vector in $\mathbb{R}^n$, as shown in Equation 1. $n = 15$ was used for the benchmarking done in this work. Vector representations of distance, such as the one used here, have been shown to be better for training neural networks than scalar distances [6]. Edges also denote whether the connected nodes are inter or intra chain nodes via a 1-hot encoding. This edge encoding was inspired in part by previous work [6].

$$\begin{gathered} f_{d\to\mathbb{R}^n} = e^{-\left(\frac{d^2}{\vec{r}}\right)} \\ \vec{r} = \{1.5^i | i = 0, 1, 2, ..., n\} \end{gathered} \tag{1}$$

The resulting graph captures the geometry of the given TCR-pMHC complex at the binding site. Importantly, this graph representation is SE(3) invariant with respect to the input complex, meaning that the output of the network will not be affected by rigid body transformations of the input TCR-pMHC complex. This invariance means that no structural alignment of our input is necessary.

We experimented with more complex node and edge features, such as the relative orientation features proposed in [6], however, we found that these additional features did not result in improved performance for our task. This may be due to such feature encodings having a higher sensitivity to noise in the models. Additionally, we chose to focus only on structural and physicochemical properties as possible node features for STAG. Not incorporating features derived from amino acid sequences allows us to better compare the merits of graph convolutional neural networks against other unimodal classifiers trained on structure data and acquire a better understanding of how to leverage structural information in TCR-pMHC binding specificity prediction.

2) Graph Convolution Operator: In this work, we make use of what we have termed the Edge-Variable-Transformer (EVT) convolution, shown in Equation 2. To our knowledge, this is the first use of this convolutional operator. Edge variable convolutions were first proposed in [50] as an effective graph convolution that provided added model interpretability. Here, we augment this convolution with an attention mechanism.

$$\begin{gathered} \mathbf{x}'_i = \mathbf{\Theta}_0\mathbf{x}_i + \sum_{j\in\mathcal{N}(i)} \alpha_{i,j}(\mathbf{x}_j \cdot h_{\mathbf{\Theta}_1}(\mathbf{e}_{ij})) \\ \alpha_{i,j} = \text{softmax}\left(\frac{(\mathbf{\Theta}_2\mathbf{x}_i)^\top \mathbf{\Theta}_3\mathbf{x}_j}{\sqrt{d}}\right) \end{gathered} \tag{2}$$

Each Θ is a learnable weight matrix while h_Θ is a Multi-Layer-Perceptron (MLP) that learns a mapping from an edge vector e_{ij} to a weight matrix. The values of each Θ and h_Θ are initialized using Xavier initialization [51] and learned through stochastic gradient descent during training. x_i is the central node and the x_j are the neighboring nodes of x_i, $\mathcal{N}(i)$. d is the dimension of the softmax output and the values passed to the softmax function are scaled by $\sqrt{d}$ for numerical stability.

3) Graph Convolutional Neural Network: The architecture of STAG is shown in Figure 3. Given an input graph, constructed as described above, an MLP consisting of linear, non-linear, and regularization layers is first applied to the nodes. Next, each edge has the values of its sender and receiver nodes concatenated to it. The edges are then passed through an MLP comprised of linear, non-linear, and regularization layers. Then, the graph is passed through three message passing graph convolutional blocks, each made up of a graph convolution, layer normalization, and a non-linearity. Finally, global mean pooling is applied to all node values and the result is passed through fully connected layers to return a binary prediction. Layer normalization and dropout are applied as regularization methods throughout the architecture to prevent over-fitting. It is worth noting here that STAG has considerably fewer layers than some other graph neural networks used in proteomics [7]. We converged on this architecture through experimentation according to empirical results.

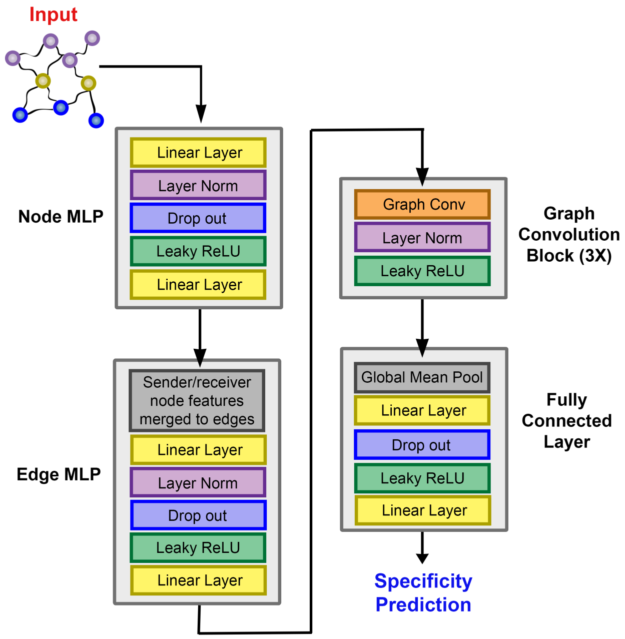

Fig. 3. A visualization of the STAG architecture.

We experimented with various convolutional operators within our architecture. All operators were implemented using the PyTorch Geometric. Their median AUCs after one round of 5-fold cross validation on our pan-peptide dataset are shown in Table I. Each of these convolutional operators makes use of both the node and edge features of our constructed TCR-pMHC graphs. We chose to benchmark two versions of STAG against the other classifiers considered in this work, one using each of the two highest performing convolutional operators, the TransformerConv and the EVT Conv.

B. Datasets

1) Peptide Specific Datasets: Classifiers trained to predict TCR binding for a specific pMHC complex have been shown to outperform classifiers trained for pan-peptide prediction [19]. Here we curate 8 multi-modal peptide specific datasets

TABLE I
MEDIAN ROC-AUC VALUES FOR THE STAG ARCHITECTURE WITH DIFFERENT GRAPH CONVOLUTIONAL OPERATORS.

Convolutional Operator	AUC
CGConv [52]	0.736
GATConv [53]	0.739
GATv2Conv [54]	0.741
NNConv [50]	0.750
TransformerConv [55]	0.766
EVTConv (ours)	**0.791**

containing amino acid sequences and protein structures; see Table II. The modeled structures in each dataset can be used to train STAG, the 3D-CNN, or other structure-based methods while the amino acid sequences can be used to train sequence methods like NetTCR 2.2 and ERGO II. These 8 peptides were selected because they are well studied and known to bind with several TCRs [56].

TABLE II
THE 8 PEPTIDE SPECIFIC DATASETS

Peptide	MHC	Dataset Size
GILGFVFTL	A*02:01	2,281
RAKFKQLL	B*08:01	1,884
ELAGIGILTV	A*02:01	1,339
KLGGALQAK	A*03:01	1,161
NLVPMVATV	A*02:01	964
AVFDRKSDAK	A*11:01	875
GLCTLVAML	A*02:01	509
IVTDFSVIK	A*11:01	335

The TCRs in the 8 datasets were sampled from public databases. Positive TCR-pMHC pairs were curated from the McPAS [57], VDJ [58], and IEDB [56] databases as well as from the 10x genomics public datasets [59]. The TCR-pMHCs taken from the VDJ database were filtered to exclude those with a confidence score equal to zero. TCR-pMHCs taken from the 10x genomics datasets were filtered according to the ITRAP algorithm [60], which has been shown to retain a diverse and large dataset while effectively removing false positives from the 10x genomics datasets [60]. No TCR-pMHC pairs were filtered out from the McPAS or IEDB databases in this step. Negative TCR-pMHC pairs were sampled from the 10x genomics datasets.

The datasets were then pruned to remove TCRs with similar CDR3α and CDR3β sequences to prevent data leakage. The CDR3α and CDR3β amino acid sequences of each data point were measured in similarity to the others using the kernel similarity metric proposed in [61]. The datasets were then pruned so that there were no decoys with kernel similarity greater than 0.90 and no ground truth samples with kernel similarity greater than 0.95. These thresholds have been set in previous work [19]. While there is an abundance of non-binding TCR-pMHC pairs in the 10x genomics datasets, we chose to maintain an approximate 4:1 ratio of negative to positive TCRs in each dataset found in Table II as was done in previous work [20], [21], [34], [36].

Once we had curated diverse datasets of TCRs for each of our 8 peptides, the amino acid sequences and predicted structures of each TCR and TCR-pMHC pair were calculated. TCRs are described in public databases using their V, D, J, and CDR3 regions. We used the tool STITCHR [62] to translate V, D, J, and CDR3 information into full amino acid sequences. The full amino acid sequences were then passed to the tools TCRpMHCmodels and immune-scape to model 3D protein structures. STITCHR is a hidden markov model that calculates full amino acid sequences for the alpha and beta chains of the TCR from V, D, J, and CDR3 information with extremely high fidelity. TCRpMHCmodels and immune-scape are tools that use homology modeling to produce approximate structures of TCRs and pMHCs. Note that neither TCRpMHCmodels nor immune-scape utilize AlphaFold2.

The end result of this curation and processing is 8 multi-modal peptide specific datasets containing thousands of wet-lab validated binding and non-binding TCRs. These datasets include TCR and pMHC amino acid sequences that can be used to train and test sequence methods such as NetTCR 2.2 and ERGO II, as well as modeled 3D structures that can be used to train and test structure methods such as STAG, the 3D-CNN, and methods based on inter-protein contacts. Such datasets are invaluable in comparing sequence and structure-based methods for TCR-pMHC binding prediction and we hope that they will be of use to future researchers.

2) Pan-Peptide Dataset: In addition to the 8 peptide specific datasets curated and considered in this work, we curated a multi-modal pan-peptide dataset to test each classifier's performance on the pan-peptide prediction task. This dataset contains $18,260$ labeled TCR-pMHC pairs with their full amino acid sequences and corresponding 3D structural models. This diverse dataset contains 330 unique peptide-MHC class I complexes and over $8,500$ unique TCRs.

Positive TCR-pMHC pairs were again curated from the McPAS, VDJ, IEDB databases and from the 10x genomics public datasets, with the same filtering techniques described above being applied to the data. Additionally, we restricted our analysis to peptides with 8-13 amino acids, as MHC class I peptides are typically within this range. We also removed all pMHCs that were not of human origin. Finally, upon observing that The A*03 and A*11 dextramers in the 10x datasets appear to have high levels of non-specific binding, we chose to exclude all positive A*11 and A*03 samples from the 10x datasets.

Negative data points were both sampled from the 10x genomics dataset and generated through swapping TCRs. When a TCR is known to bind with a given pMHC, the likelihood of the TCR binding to the new pMHC is estimated to be about 1 in 20,000 [63], meaning that randomly selecting a TCR and pairing it with a new pMHC is a valid method for generating a negative data point. These "swapped negatives" have been used in dozens of ML papers on TCR-pMHC interaction and are believed to only have a minor impact on classifier performance [19]. All datasets mentioned in this study were curated in June of 2023.

IV. RESULTS

A. Performance Evaluation on Peptide Specific Datasets

To evaluate the merits of our graph-based method for predicting TCR-pMHC binding specificity from modeled struc-

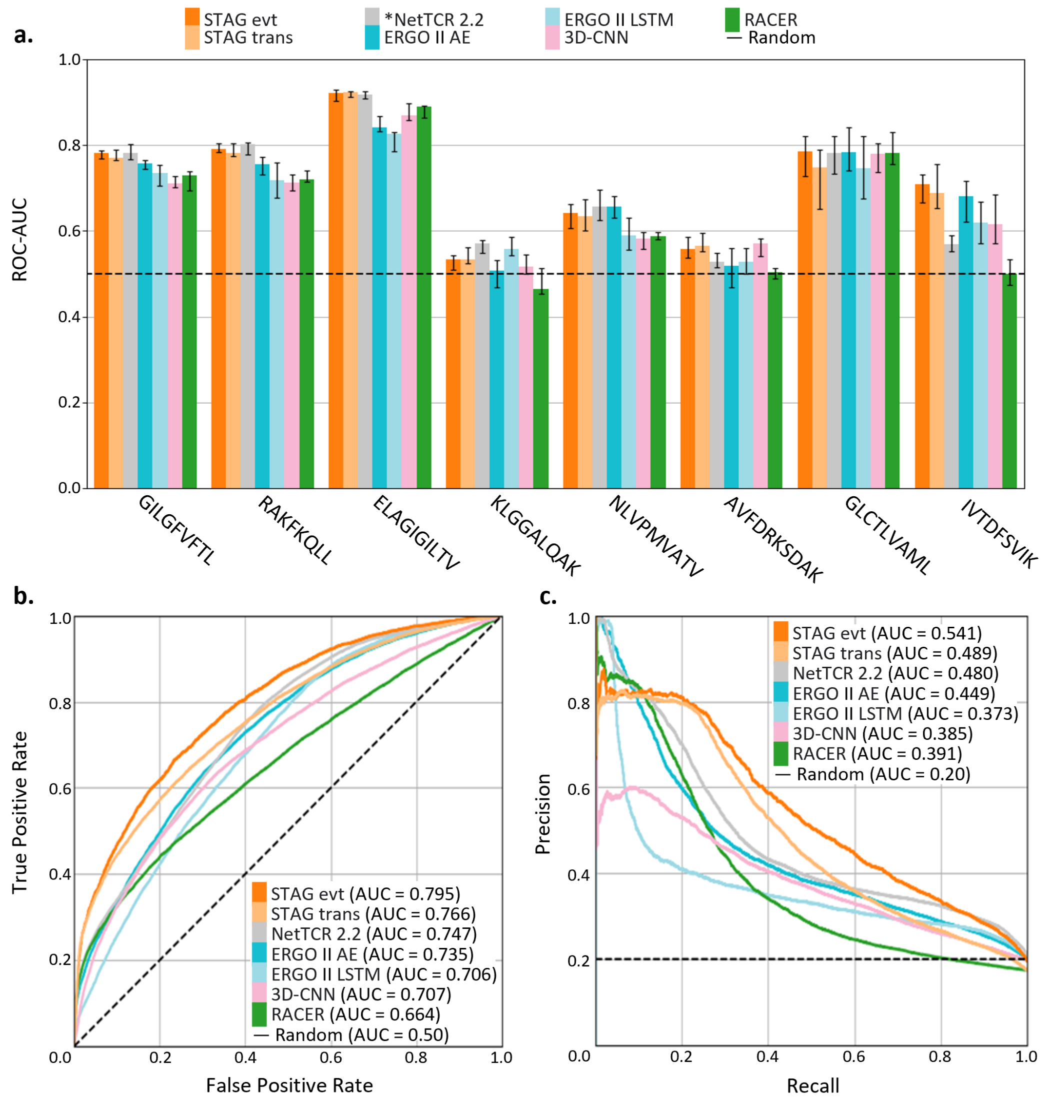

Fig. 4. **a.** Median ROC-AUC values for each of the classifiers on the 8 peptide specific datasets as measured during repeated cross validation. **NetTCR 2.2 was first trained on the pan-peptide dataset, then fine-tuned for each peptide specific dataset.* **b.** Median ROC curve for each of the classifiers on the pan-peptide dataset as measured during repeated cross validation. **c.** Median precision-recall curve for each of the classifiers on the pan-peptide dataset as measured during repeated cross validation.

ture data, we compared STAG to two other models that predict binding specificity from structure. These are the RACER protocol for specificity prediction and a 3D-CNN trained on voxel representations of the proteins. Our results show that STAG outperforms the other structure-based methods across all 8 datasets.

To compare our structure-based method to sequence-based methods, we benchmarked NetTCR 2.2 and ERGO II on all 8 of the peptide-specific datasets. All models were trained and tested on the same data and benchmarked using repeated 5-fold cross validation (see Section 1 of the Supplementary Material for details). The resulting ROC-AUC values for each model trained on protein structures and protein sequences are shown in Figure 4a (see Section 4 of the Supplementary Material for PR-AUC). STAG trans represents the STAG method with the graph-transformer convolution. STAG evt represents the STAG method with the novel EVT graph convolution. Figure 4a shows that STAG performs comparably or better than the sequence-based methods across the 8 peptide specific datasets in terms of ROC-AUC. Note that AUCs recorded here for the tools NetTCR 2.2 and ERGO II are slightly better or worse than those reported in the corresponding initial publications [20], [21]. This difference in performance reinforces the need for training TCR-pMHC binding specificity predictors on the same datasets when benchmarking them against one another, as was done in this study.

B. Performance Evaluation on Pan-Peptide Dataset

To assess each classifier's performance for the pan-peptide prediction task, we performed repeated 5-fold cross validation for each classifier on our pan-peptide dataset. Figure 4b shows the ROC-AUC curves for the different classifiers considered in this study when trained and evaluated using our pan-peptide dataset. Figure 4c shows the AUPRC values for the classifiers when trained and evaluated on the pan-peptide dataset. In the pan-peptide prediction task, both variations of STAG significantly outperform the other methods. the statistical significance of all results were estimated by comparing the distributions ROC-AUC values that resulted from the repeated cross validation with those of the other classifiers through Welch's T-tests [64]. Further information, including p-values for the results, can be found in Section 5 of the Supplementary Material.

V. Interpretability

Machine learning models, such as STAG, that make predictions based upon protein structures can provide visual intuition for their predictions. The visual intuition offers greater interpretability than methods trained on sequence alone. In this light, STAG facilitates the visualization of which amino acid contacts were most important to the model when making a prediction through analysis of the edges of the graph. In addition, STAG provides information on how model's predictions are influenced by not just the peptide-TCR contacts but also the contacts between the MHC and TCR. Methods like NetTCR 2.2 and ERGO II that do not consider amino acids in the MHC individually cannot provide such insights.

To provide an example, we use a method for interpreting the predictions given by STAG that is based on the integrated gradients algorithm [65] implemented in the captum python library [66]. Figure 5 shows the normalized values of the integrated gradients of each amino acid in the TCR-pMHC complex by color. In Figure 5 we can see that STAG places the most importance on the amino acids in the peptide that make contact with the TCR and likewise places greater importance on the amino acids in the TCR's CDR loops that come closest to the peptide. This result agrees with the current consensus about which residues are expected to play the greatest role in binding specificity between TCRs and pMHCs [8], [37]. Additional examples of integrated gradient values calculated by STAG superimposed onto input structures can be found in Section 7 of the Supplementary Material. A comparison of the feature attributions given by STAG and the sequence-based methods is given in Section 8 of the Supplementary Material.

It was recently shown that sequence-based classifiers often put emphasis on residues in the CDR3-β loop and the peptide with low proximity in the actual structures that are not likely to impact binding [10], [67]. This is problematic as it may lead to incorrect or uninterpretable predictions. The utilization of a graph neural network to represent the input in STAG allows us to derive direct insight on where the important residues are in the structure according to the model.

VI. Discussion

Accurately predicting TCR-pMHC binding specificity is one of the most significant challenges in modern immunology. Successfully doing so could revolutionize cancer treatment, drug design, and vaccine development. Here we have presented STAG, a novel approach to TCR-pMHC binding specificity prediction that uses graph neural networks to capture the geometry of TCR-pMHC complexes when predicting binding specificity. We have shown that STAG consistently outperforms other methods for predicting TCR-pMHC binding specificity from structure data and achieves comparable performance to widely used tools that make predictions from sequence data. We have also shown that STAG offers potentially high interpretability. The predictions made by STAG are explainable through the visualization of amino acids that led to the model making said predictions. This explainability in the model could directly translate to more interpretable proposal of candidates for immunotherapy treatments such as peptide vaccines and adoptive cell therapy.

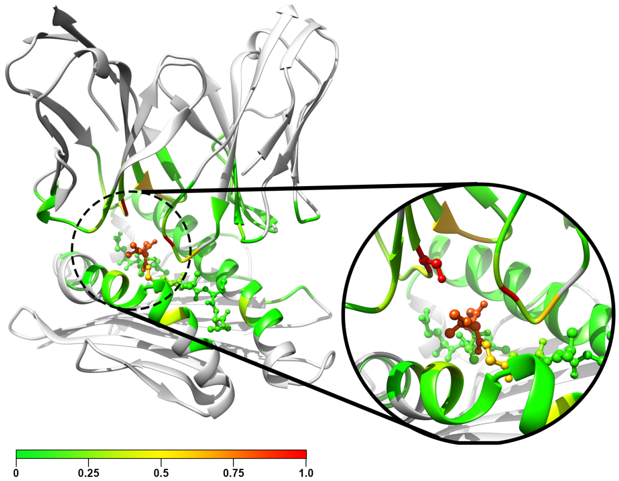

Fig. 5. A visualization of the relative importance of amino acids in the TCR-pMHC to the prediction given by STAG as calculated via the integrated gradients algorithm. Scores are normalized between zero and one with zero (green) signifying low importance and one (red) signifying high importance.

While many other ML methods have been proposed to computationally map TCRs to their cognate peptides, properly comparing them has remained a challenge. The challenge arises due to the fact that previous work often trained and tested the classifiers on distinct datasets during benchmarking. In this work, emphasis was placed on utilizing the same testing and training sets for all methods compared, giving a clearer comparison of the tools. In order to make this comparison, we curated various diverse TCR-pMHC class I binding datasets that include 3D models of the TCR-pMHC structures in addition to their amino acid sequences.

In this work, we chose to employ uni-modal models trained on either 3D protein structure data or amino acid sequences. We also forewent the use of transfer learning in order to achieve a better comparison of the effectiveness of the structure-based models tested. It has been shown for similar problems, however, that combining 3D structure with sequence data makes for better classifiers [68]. In future work we will study the application of multi-modal learning to TCR-pMHC binding prediction. We anticipate that the comparisons of different structure-based ML methods given in this work and the large multi-modal datasets published here will be of great use in this task.

All datasets used in this article and the code for inference and explainability methods published therein are available on GitHub, at https://github.com/KavrakiLab/STAG_public.

VII. Competing interests

No competing interest is declared.

VIII. Author contributions statement

J.K.S., M.M.R., and L.E.K. conceived the experiments; J.K.S. conducted the experiments; J.K.S., A.C., M.M.R., A.R., and L.E.K. wrote and reviewed the manuscript.

IX. Acknowledgments

The authors would like to thank colleagues from the Kavraki Lab and Reuben Laboratory for many helpful discussions. We also thank members of the Systems Immunology lab at RIMD, Osaka University for providing us with the resources to run their tool *immune-scape* locally. PyMOL and ChimeraX were used for visualization of structures.

X. Funding

J.K.S. is supported by a training fellowship from the Gulf Coast Consortia on the Training Program in Biomedical Informatics, National Library of Medicine 5T15LM007093. A.C. is supported by a fellowship from The Molecular Sciences Software Institute under NSF grant CHE-2136142 and by National Institutes of Health NIH U01CA258512. L.E.K. is supported in part by National Institutes of Health NIH U01CA258512. M.M.R. is supported by a CPRIT fellowship RP170593 and by National Institutes of Health NIH U01CA258512. AR is supported by the Exon 20 Group, Rexanna's Foundation for Fighting Lung Cancer, the Waun Ki Hong Lung Cancer Research Fund, MD Anderson's Lung Cancer Moon Shot, the Petrin Fund, the Salgado Family Charitable Fund, the Troper Wojcicki Foundation, an NIH/NCI R21 (FP00017376), a US Department of Defense Lung Cancer Research Program Idea Development Award (FP00017091), an AACR Career Development Award (FP00018394), the University Cancer Foundation via the Institutional Research Grant program at the University of Texas MD Anderson Cancer Center, the Happy Lungs Project, an EGFR Resisters/LUNGevity Foundation EGFR-positive research award (FF2024-00063001), a RETpositive/LUNGevity Translational Research Award Program (FF2022-00061026), a Cancer Prevention & Research Institute of Texas High Impact High Reward Award (RP210137) and a Cancer Prevention & Research Institute of Texas Individual Investigator Research Award for Computational Systems Biology of Cancer (RP230363).

References

[1] S. A. Rosenberg, N. P. Restifo, J. C. Yang, R. A. Morgan, and M. E. Dudley, "Adoptive cell transfer: a clinical path to effective cancer immunotherapy," *Nature Reviews Cancer*, vol. 8, no. 4, pp. 299–308, 2008.

[2] D. Hudson, R. A. Fernandes, M. Basham, G. Ogg, and H. Koohy, "Can we predict T cell specificity with digital biology and machine learning?," *Nature Reviews Immunology*, vol. 23, no. 8, p. 511–521, 2023.

[3] C. Soto, R. G. Bombardi, M. Kozhevnikov, R. S. Sinkovits, E. C. Chen, A. Branchizio, N. Kose, S. B. Day, M. Pilkinton, M. Gujral, *et al.*, "High frequency of shared clonotypes in human T cell receptor repertoires," *Cell reports*, vol. 32, no. 2, 2020.

[4] J. Robinson, M. J. Waller, P. Parham, J. G. Bodmer, and S. G. Marsh, "IMGT/HLA Database—a sequence database for the human major histocompatibility complex," *Nucleic Acids Research*, vol. 29, no. 1, pp. 210–213, 2001.

[5] K. Saotome, D. Dudgeon, K. Colotti, M. J. Moore, J. Jones, Y. Zhou, A. Rafique, G. D. Yancopoulos, A. J. Murphy, J. C. Lin, *et al.*, "Structural analysis of cancer-relevant tcr-cd3 and peptide-mhc complexes by cryoem," *Nature Communications*, vol. 14, no. 1, p. 2401, 2023.

[6] O.-E. Ganea, X. Huang, C. Bunne, Y. Bian, R. Barzilay, T. S. Jaakkola, and A. Krause, "Independent SE(3)-equivariant models for end-to-end rigid protein docking," in *International Conference on Learning Representations*, 2022.

[7] C. Zhao, T. Liu, and Z. Wang, "PANDA2: protein function prediction using graph neural networks," *NAR genomics and bioinformatics*, vol. 4, no. 1, p. lqac004, 2022.

[8] D. K. Sasmal, W. Feng, S. Roy, P. Leung, Y. He, C. Cai, G. Cao, H. Lian, J. Qin, E. Hui, *et al.*, "TCR–pMHC bond conformation controls TCR ligand discrimination," *Cellular & Molecular Immunology*, vol. 17, no. 3, pp. 203–217, 2020.

[9] G. Petrova, A. Ferrante, and J. Gorski, "Cross-reactivity of T cells and its role in the immune system," *Critical Reviews™ in Immunology*, vol. 32, no. 4, 2012.

[10] C. Dens, W. Bittremieux, F. Affaticati, K. Laukens, and P. Meysman, "Interpretable deep learning to uncover the molecular binding patterns determining TCR–epitope interaction predictions," *ImmunoInformatics*, p. 100027, 2023.

[11] J. Leem, S. H. P. de Oliveira, K. Krawczyk, and C. M. Deane, "STCRDab: the structural T-cell receptor database," *Nucleic acids research*, vol. 46, no. D1, pp. D406–D412, 2018.

[12] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, *et al.*, "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[13] J. Zhang, W. Ma, and H. Yao, "Accurate TCR-pMHC interaction prediction using a BERT-based transfer learning method," *Briefings in Bioinformatics*, vol. 25, no. 1, p. bbad436, 2024.

[14] S. Yadav, D. S. Vora, D. Sundar, and J. K. Dhanjal, "TCR-ESM: Employing protein language embeddings to predict TCR-peptide-MHC binding," *Computational and Structural Biotechnology Journal*, vol. 23, pp. 165–173, 2024.

[15] Y. Gao, Y. Gao, Y. Fan, C. Zhu, Z. Wei, C. Zhou, G. Chuai, Q. Chen, H. Zhang, and Q. Liu, "Pan-Peptide meta learning for T-cell receptor–antigen binding recognition," *Nature Machine Intelligence*, vol. 5, no. 3, p. 236–249, 2023.

[16] M.-D. N. Pham, T.-N. Nguyen, L. S. Tran, Q.-T. B. Nguyen, T.-P. H. Nguyen, T. M. Pham, H.-N. Nguyen, H. Giang, M.-D. Phan, and V. Nguyen, "EPITCR: A highly sensitive predictor for TCR–peptide binding," *Bioinformatics*, vol. 39, no. 5, 2023.

[17] L. Zhang, H. Li, Z. Zhang, J. Wang, G. Chen, D. Chen, W. Shi, G. Jia, and M. Liu, "Hybrid gMLP model for interaction prediction of MHC-peptide and TCR," *Frontiers in Genetics*, vol. 13, p. 1092822, 2023.

[18] P. Bradley, "Structure-based prediction of T cell receptor: peptide-MHC interactions," *Elife*, vol. 12, p. e82813, 2023.

[19] A. Montemurro, L. E. Jessen, and M. Nielsen, "NetTCR-2.1: Lessons and guidance on how to develop models for TCR specificity predictions," *Frontiers in Immunology*, vol. 13, 2022.

[20] I. Springer, N. Tickotsky, and Y. Louzoun, "Contribution of T cell receptor alpha and beta CDR3, MHC typing, V and J genes to peptide binding prediction," *Frontiers in Immunology*, vol. 12, 2021.

[21] M. F. Jensen and M. Nielsen, "Nettcr 2.2 - improved tcr specificity predictions by combining pan- and peptide-specific training strategies, loss-scaling and integration of sequence similarity," *eLife*, Oct. 2023.

[22] M. Cai, S. Bang, P. Zhang, and H. Lee, "ATM-TCR: TCR-epitope binding affinity prediction using a multi-head self-attention model," *Frontiers in Immunology*, vol. 13, 2022.

[23] K. Dhusia, Z. Su, and Y. Wu, "A structural-based machine learning method to classify binding affinities between TCR and peptide-MHC complexes," *Molecular Immunology*, vol. 139, p. 76–86, 2021.

[24] R. Ehrlich, L. Kamga, A. Gil, K. Luzuriaga, L. K. Selin, and D. Ghersi, "SWARMTCR: A computational approach to predict the specificity of T cell receptors," *BMC Bioinformatics*, vol. 22, no. 1, 2021.

[25] S. Gielis, P. Moris, W. Bittremieux, N. De Neuter, B. Ogunjimi, K. Laukens, and P. Meysman, "Detection of enriched T cell epitope specificity in full T cell receptor sequence repertoires," *Frontiers in Immunology*, vol. 10, 2019.

[26] H. Huang, C. Wang, F. Rubelt, T. J. Scriba, and M. M. Davis, "Analyzing the mycobacterium tuberculosis immune response by T-cell receptor

clustering with GLIPH2 and genome-wide antigen screening," *Nature Biotechnology*, vol. 38, no. 10, p. 1194–1202, 2020.

[27] X. Lin, J. T. George, N. P. Schafer, K. Ng Chau, M. E. Birnbaum, C. Clementi, J. N. Onuchic, and H. Levine, "Rapid assessment of T-cell receptor specificity of the immune repertoire," *Nature Computational Science*, vol. 1, no. 5, p. 362–373, 2021.

[28] T. Lu, Z. Zhang, J. Zhu, Y. Wang, P. Jiang, X. Xiao, C. Bernatchez, J. V. Heymach, D. L. Gibbons, J. Wang, and et al., "Deep learning-based prediction of the T cell receptor–antigen binding specificity," *Nature Machine Intelligence*, vol. 3, no. 10, p. 864–875, 2021.

[29] M. Milighetti, J. Shawe-Taylor, and B. Chain, "Predicting T cell receptor antigen specificity from structural features derived from homology models of receptor-peptide-major histocompatibility complexes," *Frontiers in Physiology*, vol. 12, 2021.

[30] Y. Han, Y. Yang, Y. Tian, F. J. Fattah, M. S. v. Itzstein, Y. Hu, M. Zhang, X. Kang, D. M. Yang, J. Liu, *et al.*, "pan-mhc and cross-species prediction of t cell receptor-antigen binding," *bioRxiv*, pp. 2023–12, 2023.

[31] P. Moris, J. De Pauw, A. Postovskaya, S. Gielis, N. De Neuter, W. Bittremieux, B. Ogunjimi, K. Laukens, and P. Meysman, "Current challenges for unseen-epitope TCR interaction prediction and a new perspective derived from image classification," *Briefings in Bioinformatics*, vol. 22, no. 4, 2020.

[32] J.-W. Sidhom, H. B. Larman, D. M. Pardoll, and A. S. Baras, "DeepTCR is a deep learning framework for revealing sequence concepts within T-cell repertoires," *Nature Communications*, vol. 12, no. 1, 2021.

[33] A. S. Smirnov, A. V. Rudik, D. A. Filimonov, and A. A. Lagunin, "TCR-Pred: A new web-application for prediction of epitope and MHC specificity for CDR3 TCR sequences using molecular fragment descriptors," *Immunology*, vol. 169, no. 4, p. 447–453, 2023.

[34] I. Springer, H. Besser, N. Tickotsky-Moskovitz, S. Dvorkin, and Y. Louzoun, "Prediction of specific TCR-peptide binding from large dictionaries of TCR-peptide pairs," *Frontiers in Immunology*, vol. 11, 2020.

[35] W. Zhang, P. G. Hawkins, J. He, N. T. Gupta, J. Liu, G. Choonoo, S. W. Jeong, C. R. Chen, A. Dhanik, M. Dillon, and et al., "A framework for highly multiplexed dextramer mapping and prediction of T cell receptor sequences to antigen specificity," *Science Advances*, vol. 7, no. 20, 2021.

[36] V. I. Jurtz, L. E. Jessen, A. K. Bentzen, M. C. Jespersen, S. Mahajan, R. Vita, K. K. Jensen, P. Marcatili, S. R. Hadrup, B. Peters, *et al.*, "NetTCR: sequence-based prediction of TCR binding to peptide-MHC complexes using convolutional neural networks," *BioRxiv*, p. 433706, 2018.

[37] A. F. Fonseca and D. A. Antunes, "CrossDome: an interactive R package to predict cross-reactivity risk using immunopeptidomics databases," *Frontiers in Immunology*, vol. 14, p. 1142573, 2023.

[38] K. K. Jensen, V. Rantos, E. C. Jappe, T. H. Olsen, M. C. Jespersen, V. Jurtz, L. E. Jessen, E. Lanzarotti, S. Mahajan, B. Peters, and et al., "TCRPMHCMODELS: Structural modelling of TCR-PMHC Class I complexes," *Scientific Reports*, vol. 9, no. 1, 2019.

[39] T. Lusiany, Z. Xu, D. S. Saputri, H. S. Ismanto, S. A. Nazlica, and D. M. Standley, "Structural modeling of adaptive immune responses to infection," *Computer-Aided Antibody Design*, p. 283–294, 2022.

[40] R. Yin, H. V. Ribeiro-Filho, V. Lin, R. Gowthaman, M. Cheung, and B.-G. Pierce, "TCRMODEL2: High-resolution modeling of T-cell receptor recognition using Deep Learning," *Nucleic Acids Research*, vol. 51, no. W1, 2023.

[41] J. Wang and N. V. Dokholyan, "MedusaDock 2.0: Efficient and accurate protein–ligand docking with constraints," *Journal of chemical information and modeling*, vol. 59, no. 6, pp. 2509–2515, 2019.

[42] X. Zhang, H. Gao, H. Wang, Z. Chen, Z. Zhang, X. Chen, Y. Li, Y. Qi, and R. Wang, "PLANET: A Multi-objective Graph Neural Network Model for Protein–Ligand Binding Affinity Prediction," *Journal of Chemical Information and Modeling*, 2023.

[43] I. Kozlovskii and P. Popov, "Protein–peptide binding site detection using 3D convolutional neural networks," *Journal of chemical information and modeling*, vol. 61, no. 8, pp. 3814–3823, 2021.

[44] S. Park and C. Seok, "GalaxyWater-CNN: Prediction of water positions on the protein structure by a 3D-convolutional neural network," *Journal of Chemical Information and Modeling*, vol. 62, no. 13, pp. 3157–3168, 2022.

[45] Y. Wang, Z. Wei, and L. Xi, "Sfcnn: a novel scoring function based on 3D convolutional neural network for accurate and stable protein–ligand affinity prediction," *BMC bioinformatics*, vol. 23, no. 1, p. 222, 2022.

[46] L. Deng, C. Ly, S. Abdollahi, Y. Zhao, I. Prinz, and S. Bonn, "Performance comparison of TCR-pMHC prediction tools reveals a strong data dependency," *Frontiers in Immunology*, vol. 14, p. 1128326, 2023.

[47] K. Jha, S. Saha, and H. Singh, "Prediction of protein–protein interaction using graph neural networks," *Scientific Reports*, vol. 12, no. 1, p. 8360, 2022.

[48] W. R. Atchley, J. Zhao, A. D. Fernandes, and T. Drüke, "Solving the protein sequence metric problem," *Proceedings of the National Academy of Sciences*, vol. 102, no. 18, pp. 6395–6400, 2005.

[49] A. Kidera, Y. Konishi, M. Oka, T. Ooi, and H. A. Scheraga, "Statistical analysis of the physical properties of the 20 naturally occurring amino acids," *Journal of Protein Chemistry*, vol. 4, pp. 23–55, 1985.

[50] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International conference on machine learning*, pp. 1263–1272, PMLR, 2017.

[51] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, JMLR Workshop and Conference Proceedings, 2010.

[52] T. Xie and J. C. Grossman, "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties," *Physical review letters*, vol. 120, no. 14, p. 145301, 2018.

[53] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *International Conference on Learning Representations*, 2018.

[54] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?," in *International Conference on Learning Representations*, 2022.

[55] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, "Masked label prediction: Unified message passing model for semi-supervised classification," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21* (Z.-H. Zhou, ed.), pp. 1548–1554, International Joint Conferences on Artificial Intelligence Organization, 8 2021. Main Track.

[56] R. Vita, J. A. Overton, J. A. Greenbaum, J. Ponomarenko, J. D. Clark, J. R. Cantrell, D. K. Wheeler, J. L. Gabbard, D. Hix, A. Sette, *et al.*, "The immune epitope database (IEDB) 3.0," *Nucleic acids research*, vol. 43, no. D1, pp. D405–D412, 2015.

[57] N. Tickotsky, T. Sagiv, J. Prilusky, E. Shifrut, and N. Friedman, "McPAS-TCR: a manually curated catalogue of pathology-associated T cell receptor sequences," *Bioinformatics*, vol. 33, no. 18, pp. 2924–2929, 2017.

[58] M. Shugay, D. V. Bagaev, I. V. Zvyagin, R. M. Vroomans, J. C. Crawford, G. Dolton, E. A. Komech, A. L. Sycheva, A. E. Koneva, E. S. Egorov, *et al.*, "VDJdb: a curated database of T-cell receptor sequences with known antigen specificity," *Nucleic acids research*, vol. 46, no. D1, pp. D419–D427, 2018.

[59] 10x Genomics, "A new way of exploring immunity–linking highly multiplexed antigen recognition to immune repertoire and phenotype," *Tech. rep*, 2019.

[60] A. Montemurro, H. R. Povlsen, L. E. Jessen, and M. Nielsen, "Benchmarking data-driven filtering for denoising of TCRpMHC single-cell data," *Scientific Reports*, vol. 13, no. 1, p. 16147, 2023.

[61] W.-J. Shen, H.-S. Wong, Q.-W. Xiao, X. Guo, and S. Smale, "Towards a mathematical foundation of immunology and amino acid chains," *arXiv preprint arXiv:1205.6031*, 2012.

[62] J. M. Heather, M. J. Spindler, M. H. Alonso, Y. I. Shui, D. G. Millar, D. S. Johnson, M. Cobbold, and A. N. Hata, "Stitchr: stitching coding TCR nucleotide sequences from V/J/CDR3 information," *Nucleic Acids Research*, vol. 50, no. 12, pp. e68–e68, 2022.

[63] J. Ishizuka, K. Grebe, E. Shenderov, B. Peters, Q. Chen, Y. Peng, L. Wang, T. Dong, V. Pasquetto, C. Oseroff, *et al.*, "Quantitating T cell cross-reactivity for unrelated peptide antigens," *The Journal of Immunology*, vol. 183, no. 7, pp. 4337–4345, 2009.

[64] B. L. Welch, "The significance of the difference between two means when the population variances are unequal," *Biometrika*, vol. 29, no. 3/4, pp. 350–362, 1938.

[65] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*, pp. 3319–3328, PMLR, 2017.

[66] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, *et al.*, "Captum: A unified and generic model interpretability library for pytorch," *arXiv preprint arXiv:2009.07896*, 2020.

[67] D. A. Antunes, M. M. Rigo, M. V. Freitas, M. F. Mendes, M. Sinigaglia, G. Lizée, L. E. Kavraki, L. K. Selin, M. Cornberg, and G. F. Vieira, "Interpreting T-cell cross-reactivity through structure: implications for TCR-based cancer immunotherapy," *Frontiers in immunology*, vol. 8, p. 1210, 2017.

[68] F. Wu, L. Wu, D. Radev, J. Xu, and S. Z. Li, “Integration of pretrained protein language models into geometric deep learning networks,” *Communications Biology*, vol. 6, no. 1, p. 876, 2023.