Treating biomolecular interaction as an image classification problem – a case study on T-cell receptor-epitope recognition prediction

Pieter Moris[1,2,✉], **Joey De Pauw**[1], **Anna Postovskaya**[1,2], **Benson Ogunjimi**[3,4,5,6], **Kris Laukens**[1,2], **and Pieter Meysman**[1,2]

[1]Adrem Data Lab, Department of Mathematics and Computer Science, University of Antwerp, Antwerp, 2020, Belgium
[2]Biomedical Informatics Research Network Antwerp (Biomina), University of Antwerp, Antwerp, 2020, Belgium
[3]AUDACIS, Antwerp Unit for Data Analysis and Computation in Immunology and Sequencing, University of Antwerp, Antwerp, Belgium
[4]Antwerp Center for Translational Immunology and Virology (ACTIV), Vaccine & Infectious Disease Institute (VAXINFECTIO), University of Antwerp, Antwerp, Belgium
[5]Centre for Health Economics Research & Modeling Infectious Diseases (CHERMID), Vaccine & Infectious Disease Institute (VAXINFECTIO), University of Antwerp, Antwerp, Belgium
[6]Department of Paediatrics, Antwerp University Hospital, Edegem, Belgium

The prediction of epitope recognition by T-cell receptors (TCR) has seen much progress in recent years, with several methods now available that can predict TCR-epitope recognition for a given set of epitopes. However the generic case of evaluating all possible TCR-epitope pairs remains challenging, mainly due to the high diversity of sequences and the limited amount of currently available training data. In this work, we present a novel feature engineering approach for sequence-based predictive molecular interaction models, and demonstrate its potential in generic TCR-epitope recognition. The approach is based on the pairwise combination of the physicochemical properties of the individual amino acids in both sequences. By providing a convolutional neural network (CNN) with a combined representation of the sequences, the model is primed to learn from the molecular binding context directly. It forgoes the need of separate feature extraction steps that operate on, and have to be learned for, both interactors independently. We found indications that this simplifies the prediction task and that it can improve the generalization capabilities of the model to a certain degree. We postulate that similar feature engineering methods could pave the way towards general epitope-agnostic models, although further improvements and additional data are still necessary. Note that this representation is generic and could be applied to a wide variety of similar prediction tasks where sequence-based inputs are the norm. In addition, we highlight that appropriate validation strategies are required to accurately assess the generalization performance of TCR-epitope recognition models when applied to both known and novel epitopes.

T cell epitope prediction | T cell receptor | epitope specificity | immunoinformatics | convolutional neural network | deep learning

Correspondence: *pieter.moris@uantwerpen.be*

Introduction

The main goal of the immune system is to distinguish self from non-self, as the latter is likely to be harmful to the host and perceived as a threat. Whereas the innate branch of the immune system has evolved to recognize general patterns that are conserved across pathogens, the adaptive branch is characterized by its specificity and malleability. The adaptive immunity changes continuously throughout the life of an individual; it is shaped by the threats it encounters, and builds up a memory to facilitate the mounting of defense mechanisms against similar threats in the future

For an adaptive immune response to be initiated, a T-cell receptor (TCR) on the surface of a T-cell has to recognize an immunogenic non-self or altered-self peptide (epitope), as it is presented in the context of a major histocompatibility complex (MHC) molecule on the surface of another cell. This system has to be able to recognise a large variety of epitopes, which can be derived from rapidly evolving pathogens or malignant cells locked in an evolutionary arms race. The exhaustiveness of the immune system is achieved through the combination of the highly polymorphic MHC genes and the somatic V(D)J recombination underlying the generation of T-cell receptors.

Knowing the exact patterns that govern successful TCR-pMHC recognition or being able to predict the interaction of any given TCR-epitope pair, would be an important step forward for personalized healthcare. Once the immune repertoire of a person can be analyzed and interpreted, it will become possible to estimate an individual infection progression, to guide the choice of vaccine components, to adjust hematopoietic transplants and cancer cellular immunotherapy for more individually-tailored treatments with reduced side-effects, and to help us understand a person's susceptibility to auto-immune disorders.

Given the high intra- and inter-individual diversity of the TCR-pMHC interactions and the difficulty of mapping them through experimental means, computational methods offer a promising complementary approach to gain insight into the general patterns of interaction-driven activation.

Thus far, research has primarily focused on modeling the MHC-epitope binding affinity and this has led to several reliable, accurate and precise predictive tools (1–7). However, understanding the molecular underpinnings of TCR-epitope recognition has proven to be more challenging. This can be explained in part by the lack of sufficient amounts of high quality data, although more and more experimental data is now being generated and being stored in public databases such as VDJdb and IEDB (8, 9). In addition, the large diversity of TCR and epitope sequences leads to a large, high-dimensional and likely non-homogeneous search space. This

problem needs sophisticated models that can efficiently extract the underlying distinguishing features from these highly variable molecular sequences. In essence, these models need to be able to understand the processes that govern the affinity between a TCR and epitope on a molecular level. This process is rather complex, because while sequence similarity seems to be related to the binding process, epitopes can be bound by varying TCRs (10–12) and TCRs can also display cross-reactivity (13). Despite the advent of single cell TCR paired chain sequencing, currently available TCR-epitope binding data mostly consists of single-chain, often beta-chain, data, while in reality both the alpha- and beta-chains are thought to contribute to binding specificity. Indeed, recent studies have concluded that utilising both chains could increase the accuracy of predictive TCR-epitope models (14, 15).

Previous work has tackled this problem from an epitope-specific angle, and demonstrated that the amino acid sequences of the TCR CDR3 region contain sufficient information to predict epitope recognition using epitope-specific models (10, 11, 14, 16–18). These types of predictive models have now been made accessible to immunology researchers via web tools such as TCRex (19). One shortcoming of an epitope-specific approach is that a unique model needs to be trained for every epitope (or for a set of epitopes in the case of Bi et al. (17)), which requires sufficient training examples of TCRs with the same epitope specificity.

Generic sequence-based models that can predict the interaction between any (unseen) TCR and epitope are the next frontier. However, recent studies have shown that while generic TCR-epitope prediction models can achieve a reasonable performance on interactions that involve epitopes already encountered training, they cannot yet reliable extrapolate to novel epitopes (i.e. *epitope-agnostic* models) (15, 20). This is likely caused by the large heterogeneity in potential TCR and epitope sequences, and the limited amount of readily available training data that covers this sequence space. This problem of predictive models under-performing on new data due to a shift in the distribution of the inputs (i.e. a divergent set of TCRs and/or epitopes) is more generally referred to as covariate shift (21). Moreover, it is crucial that appropriate methods are used to assess these different types of generalization performance, because they are inherently asking different questions: *"How well can we predict interactions involving known epitopes?"* versus *"How well can we predict recognition of unseen epitopes?"*

Given the fact that detecting TCR-epitope recognition can be considered more complex than MHC epitope recognition, at least in part due to the larger diversity in sequence pairing and the limited amount of available training data, training complexity constitutes even more of a limiting factor in this setting. This necessitates relevant feature encoding. Indeed, studies have shown that feature engineering is essential for the prediction of paratope-epitope binding (22) and that it can also improve the performance of models for more general bioinformatics problems (23). Moreover, when complex machine learning methods are applied to small datasets, the dangers of overfitting and memorization of examples pose an even bigger threat. In the case of TCR-epitope recognition, the potential of models to overfit on common epitope motifs should be assessed and reduced.

To tackle these issues, we present an approach to predict TCR-epitope recognition that was inspired by image classification tasks, where deep learning models have shown strong performance (24). Our method relies on a novel feature engineering method that converts the molecular sequences into an *interaction map*, which combines the physicochemical properties of both interactors on the amino-acid level. This preprocessing step avoids the creation of an internal representation for each molecule independently. This allows the model to focus more of its resources on the task of predicting the molecular binding process, by focusing immediately on an encoding of the binding context itself. Since this feature encoding sits closer to the final prediction task, overall efficiency can be expected to improve.

Using this type of interaction map as the model input differs from the prevailing approach seen in sequence-based prediction models, not only in the field of TCR-epitope prediction, but also for the modeling of other molecular interactions. The interacting molecules are usually supplied to a model as separate or concatenated inputs (1, 3, 5, 6, 15, 18, 20, 25, 26). Those types of models need to learn an internal representation for each molecule separately, before being combined again in deeper layers. Some previous approaches have also proposed a map-like representation for the interacting molecules (7), but kept a clear separation between the two interactors since they are stored in different channels of the map. Commonly, each interactor is supplied to the network as a distinct entity (i.e. in a separate input layer, spread out over different channels or as a concatenated encoding), which are then merged by the later neural network architecture. Consequently, the learning task not only consists of finding an internal representation that captures the properties of the amino acid residues in the two sequences individually, but also how to combine these higher-level features for the two interacting molecules in order to accurately predict their binding context.

In this paper, we trained a convolutional neural network (CNN) to predict the recognition of an epitope by a TCR, based on the amino acid sequences of the TCR's CDR3 region and the epitope. First, the sequences are converted into an interaction map that represents a number of physicochemical properties of each amino acid pairing between the TCR and epitope. This interaction map is then fed into the CNN, which is tasked with learning how to extract the underlying features that govern the molecular binding process. We also contrast different training schemes and types of negative data, and discuss how they estimate different types of generalization performances.

Our results indicate that relying on feature learning for both sequences individually, can restrict the model's capacity to learn, possibly through obfuscation and dilution of the molecular binding context, when compared to an input feature encoding that represents a combination of both sequences directly.

Methods

Method motivation. In brief, we have created an input encoding that more closely corresponds to the molecular binding process that dictates TCR-epitope recognition, which allowed us to reduce the complexity of the model by removing the separate embedding steps for the two molecular interactors. This in turn should aid in avoiding overfitting (i.e. memorization on the epitope level) and improves the generalization capabilities of the model.

Deep neural networks excel at extracting relevant information from complex input features and encode it in suitable intermediate representations that are relevant for meeting a certain objective, such as predicting the recognition between two molecules. Indeed, this capacity of deep neural networks to extract features from labeled training examples in a hierarchical manner, and in this manner build up a sequence of abstractions that grow in complexity throughout the network, allows them to, in theory, approximate any function (27), and is what makes them excel at tasks such as image recognition and language processing.

However, the initial representation that is chosen for the input data can still have a profound impact on both the performance and learning rate of the model. Just like in any other supervised machine learning setting, using inefficient input data, either due to the presence of irrelevant noise or the absence of a true signal, will reduce the model's capacity to learn an adequate function for the given problem, or worse, cause it to overfit to the noise instead of the true signal (i.e. the model will find itself being drawn to local (sub)optima). This is especially important when training data is limited. Similarly, highly parameterized, complex models also lead to a higher risk of overfitting.

In order to guard against these issues, even in the case of deep neural networks with their strong feature extraction capabilities, an adequate and appropriate input encoding should be chosen to maximise the model's learning potential, and a balance must be struck between reducing noise and retaining sufficient discriminating signal.

In the case of molecular interaction prediction models, the most commonly used input representation generally display a clear initial separation of the two molecular sequences, and an embedding needs to be learned for both. We propose that this type of representation could limit the model's performance, and that a different input representation of the interactors could give the models an initial nudge in the right direction.

To this end, we have designed a novel feature encoding for sequence-based input data that summarises the binding context of the two interacting partners on the molecular level. The model no longer has to learn a representation for the combined interactors on its own, model complexity can be reduced to combat overfitting. As previously stated, this is crucial for the TCR-epitope problem due to the lack of available training data and the enormous space of potential interacting molecules.

This encoding method can be thought of as the creation of an image or pixel map, and results in a 3-dimensional tensor that we will refer to as an interaction map. By lining up the sequence of the TCR and the epitope on the axes of a two-dimensional matrix, every position within it corresponds to the pairwise combination (e.g. absolute difference) of a specific physicochemical property (or other numerical property) of the amino acids at that index of the two sequences. The map can have several channels, each corresponding to a different property, analogous to the colour channels of an RGB image. An example of an interaction map of a CDR3 and epitope sequence for four physicochemical properties is given in Figure 1.

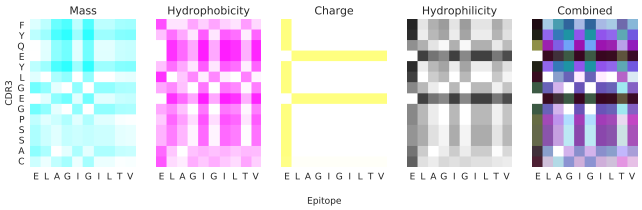

Fig. 1. An *interaction map* of a CDR3 and epitope sequence encoded by four physicochemical properties using the absolute difference as the pairwise operator. The rightmost image is a combined representation of the four two-dimensional maps, using CMYK color encoding. In the CNN, the interaction map is supplied as a three-dimensional tensor instead.

Framing TCR-epitope recognition as an image classification problem bears conceptual similarity to the familiar dot-plot method that was historically used for sequence alignment and comparison (28). In the case of interaction maps, we are interested in finding specific regions that correspond to zones that are conducive to the formation of stable molecular interactions. Multiple such hot spots can exist and contribute together to the overall affinity. Moreover, these zones consist of a combination of interacting physicochemical features. They can lie along the diagonal, or be off-set from it to represent a deviation from a linear interaction. Indeed, primary sequences remain a linear abstraction of a more complex three-dimensional structure, governed by the folding of the peptide, and distances inside the sequence will not necessarily reflect the spatial distance in a folded state. However, due to the short nature of the molecular sequences under consideration, these effects will likely be limited. Crucially, in this representation every amino acid position is contrasted with every position in the interacting partner, and no directionality is imposed on the sequences.

The motivation is that when these interaction maps are fed into a CNN, tasked with classifying two input sequences as interacting or not-interacting, the learning rate will be greater than when the two sequences are supplied separately, since the molecular binding context is explicitly provided as the starting point. An embedding can then be learned from this combined representation, rather than for both interacting sequences separately.

While convolutional layers have been used in a variety of contexts, their use seems especially appropriate for our interaction map. Indeed, one advantage of convolutional layers is that they reduce the amount of model parameters by applying the same filter to all areas of the input image (i.e. interaction map), while preserving the spatial relationship between pixels (i.e. amino acids). The underlying assumption is that any rele-

vant (low-level)feature that can be learned, should be relevant no matter its position. This assumption is also applicable to the conceptual idea behind the interaction map: regions of high affinity at the amino acid level should behave similarly regardless of their exact position within the map.

While searching for stretches of high binding potential in interaction maps makes sense from a molecular point of view, it does require the assumption that the CDR3 and epitope exist in a fairly linear conformation and form a relatively flat contact interface. This does indeed seem to be the case for many TCR-pMHC complexes, but there are also known examples where the TCR exhibits for example a deep pocket between CDR3 loops in which a side-chain of the epitope can fit (29). In addition, longer CDR3 and epitope sequences do appear to be more bulgy than shorter ones (30). However, it is also conceivable that in order to learn the signals that underlie recognition, only small stretches of the sequences need to approximately linear, an assumption that has been used in other studies as well (31). Lastly, since the interaction map is eventually fed into a CNN, it should be possible to learn more complex patterns, rather than only linear stretches, that can contribute to the binding event.

Data. The data used in this study was collected from the August 2019 release of VDJdb (8). It consists of 75,474 curated pairs of TCR CDR3 alpha/beta sequences and their epitope targets, covering both MHC classes and three species (*Homo sapiens*, *Mus musculus* and *Macaca Mulatta*).

This dataset was reduced to 20,646 unique CDR3-epitope pairs by selecting only human entries (68,506), filtering out all CDR3 sequences that were non-canonical or had unmapped V/J segments (66,597), omitting all entries originating from the 10x Genomics demonstration study (32) (26,724), removing any duplicate sequence pairs (21,951) and finally constraining the lengths of the CDR3 and epitope sequence to lie between 10-20 and 8-13 amino acids respectively. A second dataset was obtained by extracting only those pairs with a CDR3 beta-chain (TRB), for a total of 14,725 entries. The first dataset will be referred to as the *mixed-chain* dataset, the second one is as the *beta-chain* dataset.

The datasets contain 140 and 138 unique epitopes respectively, although their abundance is skewed; the majority of epitopes occur only in a few entries, whereas a small number of them appear in hundreds or thousands. An overview of the distribution of these sequences is given in Figure 2 (see Supplemental Figure S1 for a more detailed overview).

The size constraint of the sequences was set in order to limit the amount of zero-padding required for our feature encoding, and consequently limit overall model complexity. These lengths were selected in such a way to minimize the amount of data loss; about 6% of the entries in the already filtered datasets was dropped. We chose to exclude the 10x Genomics entries because several epitopes had a large amount of matching TCR sequences derived from very few individuals, which could bias our model (e.g. a singular epitope made up over 35% of the dataset before this filter). Since our model only requires the CDR3 and epitope sequence as its inputs, all other metadata that is available in VDJdb was removed (e.g. V/J segments/genes, MHC class, etc.). Lastly, we did not perform any filtering on the VDJdb confidence score.

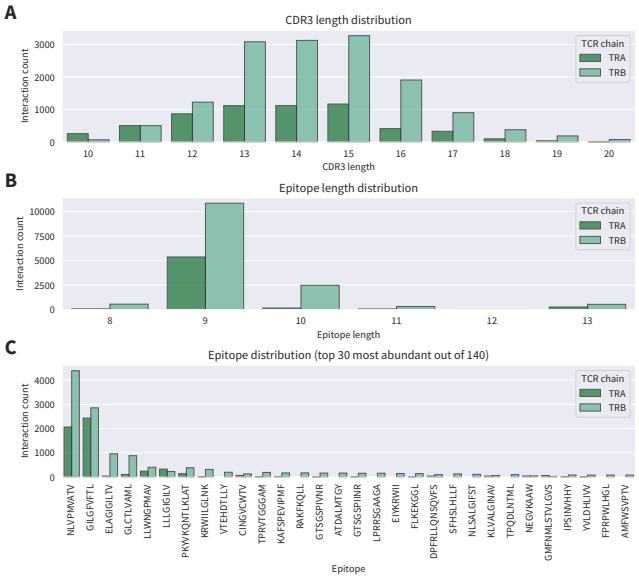

Fig. 2. Overview of contents of the filtered VDJdb data. A. Length distribution of TCR CDR3 sequence occurences per TCR chain. B. Length distribution of epitope sequence occurences per TCR chain. C. Top 20 most frequent epitopes.

An independent dataset consisting of 500,000 TRB CDR3 sequences by Dean et al. (33) was used as a reference set to generate negative examples in some experiments (see section on model training). The same size constraints as before were used and duplicate sequences were once again removed, resulting in 484,914 unique CDR3 sequences. This negative reference dataset showed minimal overlap with our filtered VDJdb dataset: only 1,179 CDR3 sequences occurred in both.

Interaction map feature encoding. The interaction map for each CDR3-epitope pair was generated by creating a vector containing physicochemical properties of the amino acids in each sequence, and then computing a matrix containing the pairwise absolute differences between the elements of the two vectors. One such two-dimensional matrix was created for each of the following physicochemical properties: charge (*pyteomics* (34, 35), hydrophobicity, hydrophilicity, mass and isoelectric point (*Biopython* (36)). Every element in the matrix was rescaled between 0 and 1, based on the maximum and minimum values a certain property displays for all twenty possible amino acids. Next, the matrices were zero-padded to a dimension of 20×13. In the end, the matrices were combined into a three-dimensional tensor (the *interaction map*), where the length and width correspond to the sequences and the depth or channels to the properties.

Network architecture. The architecture of the *interaction map CNN* is depicted in Figure 3 and follows the general design of convolutional networks for image classification. The input layer accepts interaction maps of dimensionality 20x13x5, where the first and second dimensions correspond to the amino acids of the TCR CDR3 and epitope sequences respectively, and the third (depth) corresponds to the different

physicochemical properties. Next, there is the feature extraction part, which is made up of two similar units containing a series of 2D convolutional layers followed by a 2D max pool layer, followed by dropout to reduce overfitting (37). These layers are interspersed by batch normalization steps to improve learning rates and introduce additional regularization (38), one after the first convolutional layer of each unit and one after every dropout layer. The high-level features extracted by the previous layers are then flattened and fed into a fully connected layer which takes care of the classification task; i.e. predicting whether the two inputs will bind or not.
The neurons in the convolutional layers all use rectified linear unit (ReLU) activation functions, whereas the dense layers uses hyperbolic tangent (tanh) activation and the output neuron activates based on a sigmoid function. The convolutional layers use 3x3 filters with a stride of 1 and zero-padding (to keep the height and width unchanged), and the max pool layers use a 2x2 filter to downsample and introduce some translational invariance.
For comparison, we also created a second network that represents the classical approach of separating the input of the interacting sequences, hereafter referred to as the *dual input* model) (Figure 4). Briefly, features are extracted independently from both inputs, before being concatenated and processed by deeper layers. Both inputs have only two dimensions, the amino acid sequence as the length and the BLOSUM50 encoding as the depth (39). The inputs are first passed through a single layer of 1D convolutions (consisting of various filter sizes ranging from 1 to 9), followed by batch normalization, a depth concatenation and another convolution (filter size 1). The two resulting layers are then concatenated along the (sequence) length dimension, which is flattened again in the subsequent global 1D max pool layer. Finally, the activations are passed through a fully connected layer with a single output neuron. ReLUs are used as the activation function throughout the network, with the exception of the output layer, which uses a sigmoid activation again.

Model training and validation. The networks were trained under two different cross-validation schemes: 1) a standard 10-fold cross-validation and 2) an epitope-grouped cross-validation with three folds. The former can be interpreted as "*predicting the recognition of known epitopes by TCRs*", since the epitopes that are encountered during testing have already been observed during training (albeit paired with different TCR sequences). The epitope-grouped cross-validation strategy ensures that the TCR-epitope pairs in different folds never share their epitope (i.e. every fold contains unique epitopes). Thus, it provides a representation of the performance of the model on unseen epitopes (*epitope-agnostic* generalization performance). In addition, it addresses the imbalance in the dataset with regards to the epitope distribution to some extent. The number of folds had to be limited to three, since that was the maximum number of folds that could be used while still keeping the fold size balanced and without creating any folds with only a single epitope.
We compared two methods for generating negative observations: shuffling positive pairs and sampling from a negative TRB CDR3 reference set. For the former, negative observations were generated by randomly sampling CDR3 and epitope sequences independently and removing known positive matches, a commonly used technique for interaction type data. By doing this we make the assumption that TCRs can only bind to their known epitope partner, and that all other combinations would not lead to a successful binding event. The negative observations were created independently within each fold in order to ensure that they only contained sequences that were present in the positive pairs. This forces the model to learn the actual interaction between positive combinations, rather than merely learning to recognise which sequences occur exclusively in the positive or negative observations (i.e. overfitting or *learning sequences by heart*). For the negative reference set approach, reference CDR3 sequences were matched with epitopes from the positive pairs (once again only with those present in a given fold). The frequency of positive and negative observations was kept equal in both scenarios (50:50 ratio).
We trained separate models for both the mixed-chain and the beta-chain datasets using these different validation and negative sampling approaches.
The RMSProp optimizer was used to update model weights with a learning rate of 0.001 (40). We chose binary cross-entropy as the objective function and a batch size of 512 based on our available computing power. All models were trained for 30 epochs, which proved sufficient by inspecting performance metric plots and when contrasted to early stopping approaches. We measured the generalization performance of our models using the mean and sample standard deviation (SD) of the accuracy, ROC AUC, sensitivity, specificity, (average) precision and recall over the held-out cross-validation folds.
We implemented and trained our models using TensorFlow 1.12.0 (41), CUDA (42) and the Python (3.6.8) packages Biopython 1.74 (36), Keras 2.2.4 (43), NumPy 1.17.3 (44), pandas 0.25.2 (45), pyteomics 4.1 (34, 35), scikit-learn 0.21.3 (46) and SciPy 1.3.1 (47), on NVIDIA Tesla P100 GPU compute cards provided by the High Performance Computing core facility CalcUA.

Results

Known-epitopes performance. To assess the performance of our models on TCR-epitope data with a similar background to the training data, we used a 10-fold cross-validation approach. Because of this, these performance measures should not be extrapolated to the prediction on novel, unseen epitopes. The validation ROC and precision-recall (PR) plots of both the interaction map and dual input CNNs, trained on the mixed-chain and beta-chain datasets, using this standard cross-validation approach is provided in Figure 5.
The performance of the interaction map approach seems to be at least on par with the standard dual input method (mean AUC $0.73 \pm 0.01SD$, $0.70 \pm 0.01SD$), and it performs substantially better when it comes to the precision and recall metrics (average precision $0.74 \pm 0.01SD$, $0.65 \pm 0.03SD$). The overall performance is also comparable to previously pro-

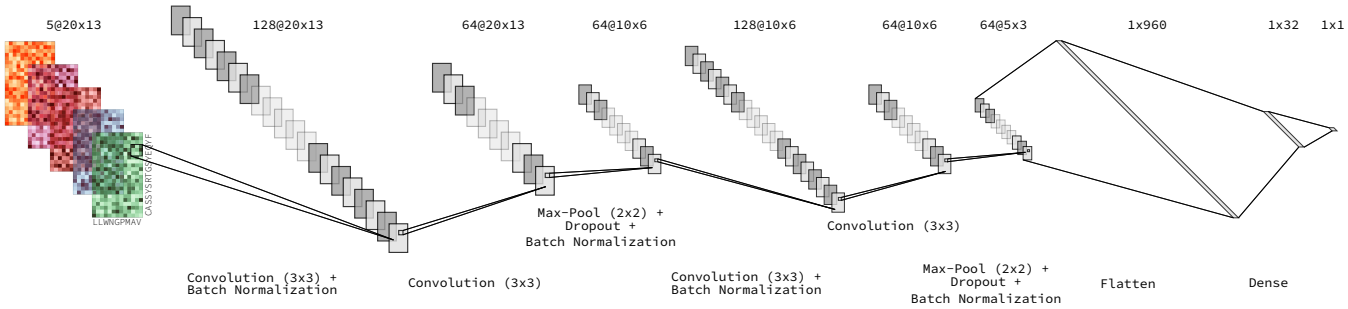

Fig. 3. Architecture of the *interaction map CNN*. The five colours in the input layer represent five different physicochemical properties used in the creation of the interaction map. Thin black lines originating from a smaller rectangle represent 2D convolutions.

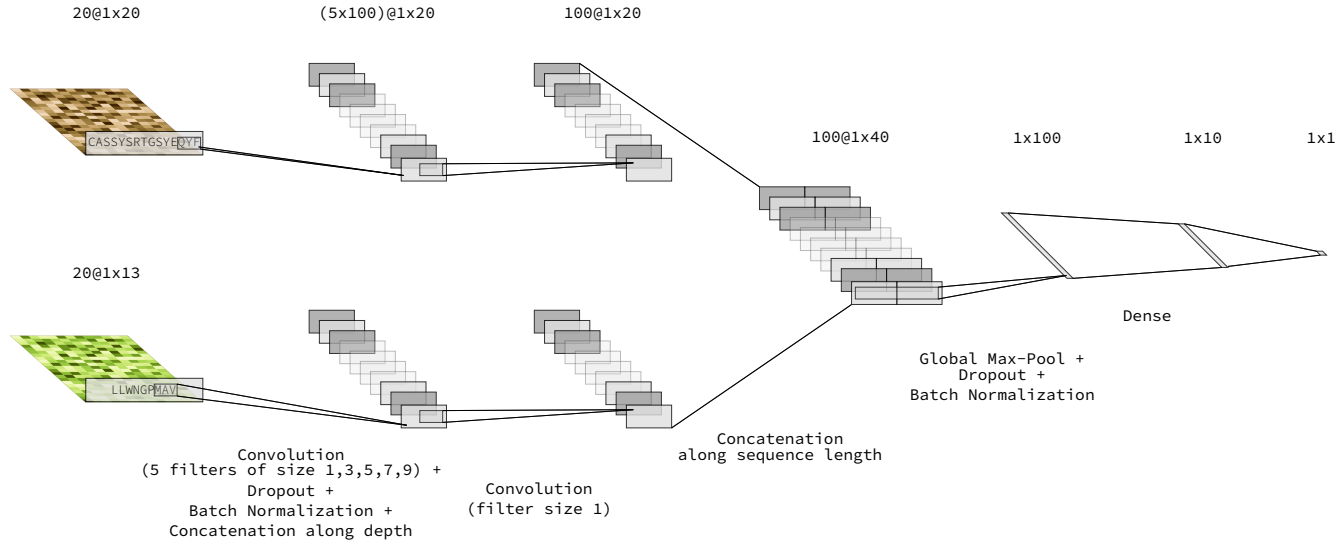

Fig. 4. Architecture of the *dual input CNN*. The two input sequences are encoded using the BLOSUM50 matrix and supplied to the model in separate layers before being combined. Note that all layers contain sequences (not images or matrices as in Figure 3), and thin black lines originating from a smaller rectangle represent 1D convolutions.

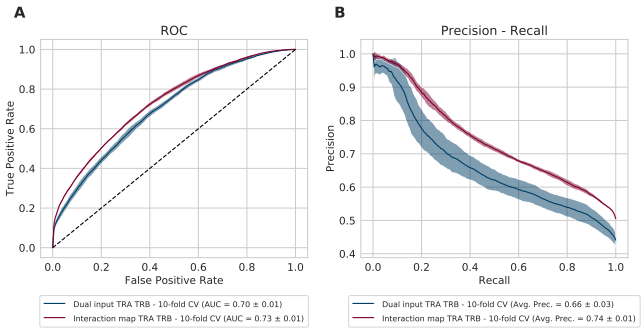

Fig. 5. ROC and PR plot for the interaction map (red) and dual input (blue) CNN after 10-fold cross-validation, when trained on the mixed-chain dataset.

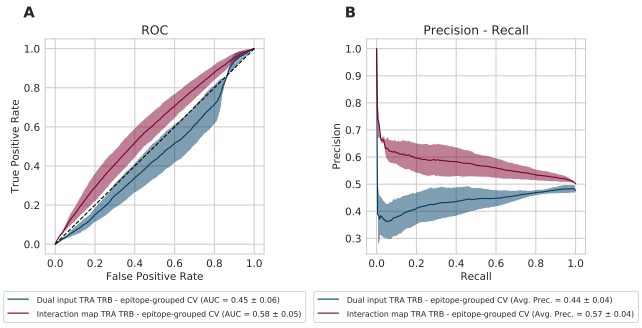

Fig. 6. ROC and PR plot for the interaction map (red) and dual input (blue) CNN after epitope-grouped cross-validation, when trained on the mixed-chain dataset.

posed general (i.e. not epitope-specific) TCR-epitope recognition models (15, 20). This suggests that the feature encoding used to create the interaction maps can provide the same information to a network as the standard approach, and even make it more tractable for the network to learn from it.

Epitope-agnostic performance. We tested the generalization to unseen epitopes via an epitope-grouped cross-validation scheme, which ensured that epitopes in the interaction pairs utilised for testing were never encountered during training. The ROC and PR plots for both CNNs are shown in Figure 6.

Both models perform poorly under this epitope-agnostic setting (mean AUC $0.58 \pm 0.05SD$, $0.45 \pm 0.06SD$; average precision $0.57 \pm 0.04SD$, $0.44 \pm 0.04SD$), which is also in line with previous studies (15, 20). However, we do find some indication that the interaction map approach is able to capture at least a small additional part of the signal in the data, in contrast to the dual input CNN.

Training with a negative reference set. We investigated the impact of different negative observations by training the interaction map CNN on the mixed-chain dataset where negative observations were either 1) generated through shuffling

(as before) or 2) derived from the negative TRB CDR3 reference dataset.

In this experiment, the model trained on the negative reference set achieved the better predictive performance (Figure 7; mean AUC $0.79 \pm 0.01SD$, $0.73 \pm 0.01SD$; average precision $0.83 \pm 0.01SD$, $0.74 \pm 0.01SD$). The crossing ROC curves do indicate that there exists a false positive threshold at which the sensitivity of the methods switches around. Similarly, the PR curve shows that the negative reference model mislabels fewer true interactions as not-binding. Nonetheless, it is still possible that this behaviour results from memorization of TCRs in negative observations, since the CDR3 sequences in negative interactions do not appear in positive ones.

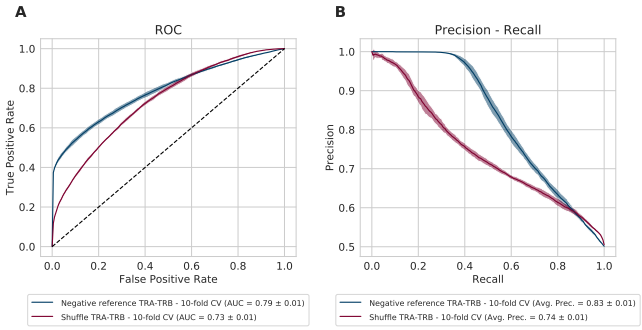

Fig. 7. ROC and PR plot for the interaction map CNN, after 10-fold cross-validation, when trained on the mixed-chain dataset with either shuffled negatives (red) or a negative reference set (blue).

Comparison of TCRα and TCRβ chains. Lastly, in terms of training on purely TRB chains or a mix of TRA and TRB, the models that were trained on mixed examples achieve the highest performance, possibly due to the larger size of the mixed dataset (20,646 versus 14,725 unique interaction pairs) (Figure 8; mean AUC $0.73 \pm 0.01SD$, $0.69 \pm 0.01SD$; average precision $0.74 \pm 0.01SD$, $0.72 \pm 0.01SD$ for mixed- and beta-chain dataset respectively). The same pattern was observed for the dual input CNN, although the evaluation metrics never superseded those of the interaction map CNN (Supplemental Figure S2), and also in the epitope-grouped setting (Supplemental Figure S3).

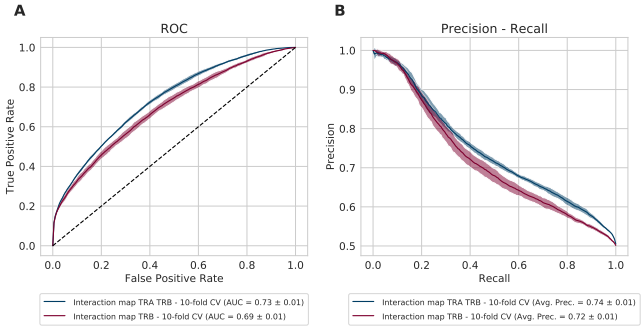

Fig. 8. ROC and PR plot for the interaction map CNN trained on the mixed-chain (blue) and beta-chain (red) dataset respectively, after 10-fold cross-validation.

Discussion

In this work, we have presented a new feature engineering method for sequenced-based interaction prediction models, based on the pairwise combination of amino acids in terms of their physicochemical properties. We have shown its potential as an input encoding for building generic, rather than epitope-specific, TCR-epitope recognition models, where it is on par or outperforms dual input embedding approaches, when its performance is not extrapolated to novel epitopes.

The lack of training examples, or conversely, the immense space covered by the human immune repertoire and the potentially interacting epitopes, remains an issue for the prediction of novel epitopes. As long as the available training data only sparsely covers the epitope sequence space, this will likely limit the predictive power of any epitope-agnostic approach. Via epitope-grouped cross-validation we have demonstrated that the *interaction map* feature representation cannot yet make this task feasible, although there is some indication that appropriate feature engineering can alleviate this problem to a small extent.

We found that models trained on CDR3-epitope pairs consisting of either TRA or TRB outperform those trained on only TRB CDR3-epitope pairs. This suggests that the types of learning methods presented here can infer the patterns that govern the binding process, since performance improves when the models are presented with more data. However, this assumes a strong similarity between the manner in which alpha- and beta-chains interact with the MHC molecule, and needs to be investigated in more detail by, for example, evaluating the performance of models on one chain after training on the other. In some sense, our results are consistent with previous research, which has shown that training on both sequences simultaneously provides better results in most cases, although this does require adequate amounts of paired data (14, 15).

Finally, our models show that the choice of negative observations has a large impact on the perceived model performance. Shuffling leads to less optimal models in general, but it is currently difficult to determine whether this behaviour will hold up in independent validation sets. Indeed, it is quite possible that the current findings result from memorization of negative observations.

Given these results, we believe there is a strong need for a more consistent and stringent testing framework for assessing the generalization performance of TCR-epitope recognition models. The influence of epitope bias is clearly shown by the difficulty of extrapolating trained models to unseen epitopes, but additional problems can crop up as well.

First of all, any variable that is correlated with TCR sequence diversity could cause issues. For example, the choice of restricting training data to a specific MHC class (even indirectly as was done here through sequence length restriction), or training on mixed species data, could introduce biases that should be assessed.

Another major issue is presented by the choice of negative data. Previous studies have already reported that validation on an independent negative set of TCRs leads to increased per-

formance (20). However, as we mentioned, this could also be the result of memorization of TCRs that are unique to the positive or negative set. This would not be an issue when shuffling TCRs, although here other biases could be introduced. For example, a negative example generated through shuffling could very well be positive in reality, and the higher the amount of similar TCRs that are present in the dataset, the higher the chance of generating such a falsely labeled observation.

Future work. More research will be required to further assess the potential of the *interaction map* encoding scheme. We have identified a list of priorities below that we would like to explore in the future.

First and foremost, we aim to further explore the impact of different training datasets, different definitions for what constitutes a proper negative sample (i.e. independent references versus shuffled combinations) and which types of validation strategies should be performed to assess epitope-agnostic performance (i.e. left-out test sets versus epitope-grouped strategies as performed in this study). The common issue underlying these problems is the limited coverage of the highly diverse potential sequence landscape of TCRs and epitopes. Consequently, these topics could be investigated in an epitope-specific manner, and by contrasting the performance for specific epitopes to the diversity it exhibits in the dataset. Related to the previous aspects, we aim to validate our approach on novel unseen epitopes obtained from new experimental data. Once these controlled benchmarks have been decided upon, our approach should be contrasted to other currently available methods in order to more accurately compare their strengths and weaknesses.

Next, the nature of the interaction map lends itself elegantly to investigating where the attention of the model is focused on. We plan to investigate this further in order to better understand which type of patterns the CNN is extracting from the maps, whether they are consistent across various TCR-epitope pairs, and whether they can be mapped onto known 3D structures of TCR-epitope complexes.

Care should also be taken to identify and minimize other potential biases in the dataset. Aside from the epitope distribution, the impact of the TCR chain, MHC class and even species bias should be investigated. This can be done through similar strategies already discussed for the epitopes, or more generally, by training models on one group and testing its performance on the other(s). In all three cases, this comes down to testing the hypothesis that the underlying molecular binding context is similar across TCR chains, MHC classes and species, and that models trained on one group can be extrapolated beyond it.

Another avenue is the extension of *interaction maps* to include additional information about the interacting molecules. For example, the inclusion of the alpha chain seems like an obvious candidate, provided that the number of paired TCR chains with known specificities continues to grow. Similarly, the other CDR loops could be included, and even the MHC context could be appended to the epitope vector. For features that consist of sequences, a choice would have to be made whether to include them via a separate input layer, or whether to append them to the *interaction map* (in case this is consistent with the spatial structure of the map and the physical molecular interaction).

Lastly, it has not escaped our notice that interaction maps could be used to model other types of molecular sequence-based interactions (e.g. protein-protein, antibody-antigen, etc.). However, there are still a number of obstacles that could make the translation to other settings difficult. Several types of molecular interactions involve longer sequences than in the TCR-epitope scenario. The length of the input sequences has a drastic impact on model complexity, since the size of the *interaction map* grows quadratically as a function of the sequence length. Moreover, the larger variance in sequence length would lead to additional zero-padding, and while CNNs are translation invariant to a certain extent, this shifting of the foci of interest across the map could still be problematic. Longer proteins can also involve stretches of amino acids which do not contribute to the molecular binding process, and are only important for scaffolding or the internal structure of a protein. Unless our method is capable of working out the folding behaviour of a sequence (which has been an open problem for decades), this will effectively amount to the dilution of the signal; only a few hot spots in the map will correspond to areas that contribute to the binding process, leaving a lot of room for noise and false positive areas in the other locations.

Conclusions

Going forward, as more TCR-epitope sequence data is being collected, we believe that appropriate feature engineering methods will be essential to maximise the potential of predictive TCR-epitope recognition models. Simultaneously, stringent and appropriate control measures will need to be devised to guide model training towards the intended goal, both in terms of what constitutes negative data and to which settings the results of different types of validation methods can be extrapolated.

ACKNOWLEDGEMENTS
We thank Danh Bui, Nicolas De Neuter, Sofie Gielis and Wout Bittremieux for all the insightful discussions. We thank all contributors to the VDJdb for making their data publicly available.

FUNDING
This work was supported by the Research Foundation – Flanders (FWO) [1141217N to PMo, 1861219N to BO] and the University of Antwerp [BOF GOA: PIMS]. The computational resources and services used in this work were provided by the HPC core facility CalcUA of the Universiteit Antwerpen, and VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government – department EWI.

AUTHOR CONTRIBUTIONS
This work was realised with the collaboration of all authors. PMo and JDP implemented the models and performed the analysis. PMe, BO and KL initiated the concept and supervised the study. All authors wrote and approved the manuscript.

COMPETING FINANCIAL INTERESTS
The authors declare no conflict of interest.

Bibliography

1. Vanessa Jurtz, Sinu Paul, Massimo Andreatta, Paolo Marcatili, Bjoern Peters, and Morten Nielsen. NetMHCpan-4.0: Improved Peptide–MHC Class I Interaction Predictions Integrating Eluted Ligand and Peptide Binding Affinity Data. *The Journal of Immunology*, 199 (9):3360–3368, November 2017. ISSN 0022-1767, 1550-6606. doi: 10.4049/jimmunol.1700893.

2. Yan Hu, Ziqiang Wang, Hailin Hu, Fangping Wan, Lin Chen, Yuanpeng Xiong, Xiaoxia Wang, Dan Zhao, Weiren Huang, and Jianyang Zeng. ACME: Pan-specific peptide–MHC class I binding prediction through attention-based deep neural networks. *Bioinformatics*, page btz427, May 2019. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btz427.
3. Haoyang Zeng and David K Gifford. DeepLigand: Accurate prediction of MHC class I ligands using peptide embedding. *Bioinformatics*, 35(14):i278–i283, July 2019. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btz330.
4. Timothy J. O'Donnell, Alex Rubinsteyn, Maria Bonsack, Angelika B. Riemer, Uri Laserson, and Jeff Hammerbacher. MHCflurry: Open-Source Class I MHC Binding Affinity Prediction. *Cell Systems*, 7(1):129–132.e4, July 2018. ISSN 24054712. doi: 10.1016/j.cels.2018.05.014.
5. Zhonghao Liu, Yuxin Cui, Zheng Xiong, Alierza Nasiri, Ansi Zhang, and Jianjun Hu. DeepSeqPan, a novel deep convolutional neural network model for pan-specific class I HLA-peptide binding affinity prediction. *Scientific Reports*, 9(1):794, December 2019. ISSN 2045-2322. doi: 10.1038/s41598-018-37214-1.
6. Poomarin Phloyphisut, Natapol Pornputtapong, Sira Sriswasdi, and Ekapol Chuangsuwanich. MHCSeqNet: A deep neural network model for universal MHC binding prediction. *BMC Bioinformatics*, 20(1):270, December 2019. ISSN 1471-2105. doi: 10.1186/s12859-019-2892-4.
7. Youngmahn Han and Dongsup Kim. Deep convolutional neural networks for pan-specific peptide-MHC class I binding prediction. *BMC Bioinformatics*, 18(1):585, December 2017. ISSN 1471-2105. doi: 10.1186/s12859-017-1997-x.
8. Dmitry V Bagaev, Renske M A Vroomans, Jerome Samir, Ulrik Stervbo, Cristina Rius, Garry Dolton, Alexander Greenshields-Watson, Meriem Attaf, Evgeny S Egorov, Ivan V Zvyagin, Nina Babel, David K Cole, Andrew J Godkin, Andrew K Sewell, Can Kesmir, Dmitriy M Chudakov, Fabio Luciani, and Mikhail Shugay. VDJdb in 2019: Database extension, new analysis infrastructure and a T-cell receptor motif compendium. *Nucleic Acids Research*, page gkz874, October 2019. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkz874.
9. Randi Vita, Swapnil Mahajan, James A. Overton, Sandeep Kumar Dhanda, Sheridan Martini, Jason R. Cantrell, Daniel K. Wheeler, Alessandro Sette, and Bjoern Peters. The Immune Epitope Database (IEDB): 2018 update. *Nucleic Acids Research*, 47(D1):D339–D343, January 2019. ISSN 1362-4962. doi: 10.1093/nar/gky1006.
10. Pradyot Dash, Andrew J. Fiore-Gartland, Tomer Hertz, George C. Wang, Shalini Sharma, Aisha Souquette, Jeremy Chase Crawford, E. Bridie Clemens, Thi H. O. Nguyen, Katherine Kedzierska, Nicole L. La Gruta, Philip Bradley, and Paul G. Thomas. Quantifiable predictive features define epitope-specific T cell receptor repertoires. *Nature*, 547(7661):89–93, July 2017. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature22383.
11. Jacob Glanville, Huang Huang, Allison Nau, Olivia Hatton, Lisa E. Wagar, Florian Rubelt, Xuhuai Ji, Arnold Han, Sheri M. Krams, Christina Pettus, Nikhil Haas, Cecilia S. Lindestam Arlehamn, Alessandro Sette, Scott D. Boyd, Thomas J. Scriba, Olivia M. Martinez, and Mark M. Davis. Identifying specificity groups in the T cell receptor repertoire. *Nature*, 547 (7661):94–98, July 2017. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature22976.
12. Pieter Meysman, Nicolas De Neuter, Sofie Gielis, Danh Bui Thi, Benson Ogunjimi, and Kris Laukens. On the viability of unsupervised T-cell receptor sequence clustering for epitope preference. *Bioinformatics*, 35(9):1461–1468, May 2019. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/bty821.
13. Galina Petrova, Andrea Ferrante, and Jack Gorski. Cross-reactivity of T cells and its role in the immune system. *Critical Reviews in Immunology*, 32(4):349–372, 2012. ISSN 1040-8401. doi: 10.1615/critrevimmunol.v32.i4.50.
14. Emmi Jokinen, Jani Huuhtanen, Satu Mustjoki, Markus Heinonen, and Harri Lähdesmäki. Determining epitope specificity of T cell receptors with TCRGP. Preprint, Bioinformatics, February 2019.
15. David S. Fischer, Yihan Wu, Benjamin Schubert, and Fabian J. Theis. Predicting antigen-specificity of single T-cells based on TCR CDR3 regions. Preprint, Bioinformatics, August 2019.
16. Nicolas De Neuter, Wout Bittremieux, Charlie Beirnaert, Bart Cuypers, Aida Mrzic, Pieter Moris, Arvid Suls, Viggo Van Tendeloo, Benson Ogunjimi, Kris Laukens, and Pieter Meysman. On the feasibility of mining CD8+ T cell receptor patterns underlying immunogenic peptide recognition. *Immunogenetics*, 70(3):159–168, March 2018. ISSN 0093-7711, 1432-1211. doi: 10.1007/s00251-017-1023-5.
17. Jingshu Bi, Yuanjie Zheng, Fang Yan, Sujuan Hou, and Chengjiang Li. Prediction of Epitope-Associated TCR by Using Network Topological Similarity Based on Deepwalk. *IEEE Access*, 7:151273–151281, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2019.2948178.
18. John-William Sidhom, H. Benjamin Larman, Drew M. Pardoll, and Alexander S. Baras. DeepTCR: A deep learning framework for revealing structural concepts within TCR Repertoire. Preprint, Bioinformatics, November 2018.
19. Sofie Gielis, Pieter Moris, Wout Bittremieux, Nicolas De Neuter, Benson Ogunjimi, Kris Laukens, and Pieter Meysman. Detection of Enriched T Cell Epitope Specificity in Full T Cell Receptor Sequence Repertoires. *Frontiers in Immunology*, 10:2820, November 2019. ISSN 1664-3224. doi: 10.3389/fimmu.2019.02820.
20. Vanessa Isabell Jurtz, Leon Eyrich Jessen, Amalie Kai Bentzen, Martin Closter Jespersen, Swapnil Mahajan, Randi Vita, Kamilla Kjærgaard Jensen, Paolo Marcatili, Sine Reker Hadrup, Bjoern Peters, and Morten Nielsen. NetTCR: Sequence-based prediction of TCR binding to peptide-MHC complexes using convolutional neural networks. Preprint, Bioinformatics, October 2018.
21. Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, October 2000. ISSN 03783758. doi: 10.1016/S0378-3758(00)00115-4.
22. Rahmad Akbar, Philippe A Robert, Milena Pavlović, Jeliazko R Jeliazkov, Igor Snapkov, Andrei Slabodkin, Cédric R. Weber, Lonneke Scheffer, Enkelejda Miho, Ingrid Hobæk Haff, Dag Trygve Tryslew Haug, Fridtjof Lund-Johansen, Yana Safonova, Geir Kjetil Sandve, and Victor Greiff. A compact vocabulary of paratope-epitope interactions enables predictability of antibody-antigen binding. Preprint, Immunology, September 2019.
23. Zheng Chen, Meng Pang, Zixin Zhao, Shuainan Li, Rui Miao, Yifan Zhang, Xiaoyue Feng, Xin Feng, Yexian Zhang, Meiyu Duan, Lan Huang, and Fengfeng Zhou. Feature selection may improve deep neural networks for the bioinformatics problems. *Bioinformatics*, page btz763, October 2019. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btz763.
24. Waseem Rawat and Zenghui Wang. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. *Neural Computation*, 29(9):2352–2449, September 2017. ISSN 0899-7667, 1530-888X. doi: 10.1162/neco_a_00990.
25. Yu Yao, Xiuquan Du, Yanyu Diao, and Huaixu Zhu. An integration of deep learning with feature embedding for protein-protein interaction prediction. *PeerJ*, 7:e7126, 2019. ISSN 2167-8359. doi: 10.7717/peerj.7126.
26. Muhao Chen, Chelsea J T Ju, Guangyu Zhou, Xuelu Chen, Tianran Zhang, Kai-Wei Chang, Carlo Zaniolo, and Wei Wang. Multifaceted protein–protein interaction prediction based on Siamese residual RCNN. *Bioinformatics*, 35(14):i305–i314, July 2019. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btz328.
27. Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, January 1993. ISSN 08936080. doi: 10.1016/S0893-6080(05)80131-5.
28. A. J. Gibbs and G. A. McIntyre. The diagram, a method for comparing sequences. Its use with amino acid and nucleotide sequences. *European Journal of Biochemistry*, 16(1):1–11, September 1970. ISSN 0014-2956. doi: 10.1111/j.1432-1033.1970.tb01046.x.
29. Jens Hennecke and Don C Wiley. T Cell Receptor–MHC Interactions up Close. *Cell*, 104 (1):1–4, January 2001. ISSN 00928674. doi: 10.1016/S0092-8674(01)00185-4.
30. Mikhail V. Pogorelyy, Alla D. Fedorova, James E. McLaren, Kristin Ladell, Dmitri V. Bagaev, Alexey V. Eliseev, Artem I. Mikelov, Anna E. Koneva, Ivan V. Zvyagin, David A. Price, Dmitry M. Chudakov, and Mikhail Shugay. Exploring the pre-immune landscape of antigen-specific T cells. *Genome Medicine*, 10(1):68, December 2018. ISSN 1756-994X. doi: 10.1186/s13073-018-0577-7.
31. Masato Ogishi and Hiroshi Yotsuyanagi. Quantitative Prediction of the Landscape of T Cell Epitope Immunogenicity in Sequence Space. *Frontiers in Immunology*, 10:827, April 2019. ISSN 1664-3224. doi: 10.3389/fimmu.2019.00827.
32. 10x Genomics. A New Way of Exploring Immunity – Linking Highly Multiplexed Antigen Recognition to Immune Repertoire and Phenotype. https://www.10xgenomics.com/resources/application-notes/a-new-way-of-exploring-immunity-linking-highly-multiplexed-antigen-recognition-to-immune-repertoire-and-phenotype/.
33. Jared Dean, Ryan O. Emerson, Marissa Vignali, Anna M. Sherwood, Mark J. Rieder, Christopher S. Carlson, and Harlan S. Robins. Annotation of pseudogenic gene segments by massively parallel sequencing of rearranged lymphocyte receptor loci. *Genome Medicine*, 7(1):123, December 2015. ISSN 1756-994X. doi: 10.1186/s13073-015-0238-z.
34. Anton A. Goloborodko, Lev I. Levitsky, Mark V. Ivanov, and Mikhail V. Gorshkov. Pyteomics—a Python Framework for Exploratory Data Analysis and Rapid Software Prototyping in Proteomics. *Journal of The American Society for Mass Spectrometry*, 24(2):301–304, February 2013. ISSN 1044-0305, 1879-1123. doi: 10.1007/s13361-012-0516-6.
35. Lev I. Levitsky, Joshua A. Klein, Mark V. Ivanov, and Mikhail V. Gorshkov. Pyteomics 4.0: Five Years of Development of a Python Proteomics Framework. *Journal of Proteome Research*, 18(2):709–714, February 2019. ISSN 1535-3893, 1535-3907. doi: 10.1021/acs.jproteome.8b00717.
36. P. J. A. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski, and M. J. L. de Hoon. Biopython: Freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11): 1422–1423, June 2009. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btp163.
37. Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.
38. Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv:1502.03167 [cs]*, March 2015.
39. Morten Nielsen, Claus Lundegaard, Peder Worning, Sanne Lise Lauemøller, Kasper Lamberth, Søren Buus, Søren Brunak, and Ole Lund. Reliable prediction of T-cell epitopes using neural networks with novel sequence representations. *Protein Science*, 12(5):1007–1017, May 2003. ISSN 09618368, 1469896X. doi: 10.1110/ps.0239403.
40. Tijmen Tieleman and Geoffrey Hinton. Rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.
41. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. 2015.
42. John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with CUDA. *Queue*, 6(2):40, March 2008. ISSN 15427730. doi: 10.1145/1365490.1365500.
43. François Chollet et al. Keras. 2015.
44. Stéfan van der Walt, S Chris Colbert, and Gaël Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, 13(2):22–30, March 2011. ISSN 1521-9615. doi: 10.1109/MCSE.2011.37.
45. Wes McKinney. Data structures for statistical computing in python. In Stéfan van der Walt and Jarrod Millman, editors, *Proceedings of the 9th Python in Science Conference*, pages 51 – 56, 2010.
46. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct): 2825–2830, 2011.
47. Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W.

Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1 0 Contributors. SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python. *arXiv:1907.10121 [physics]*, July 2019.

Supplementary information

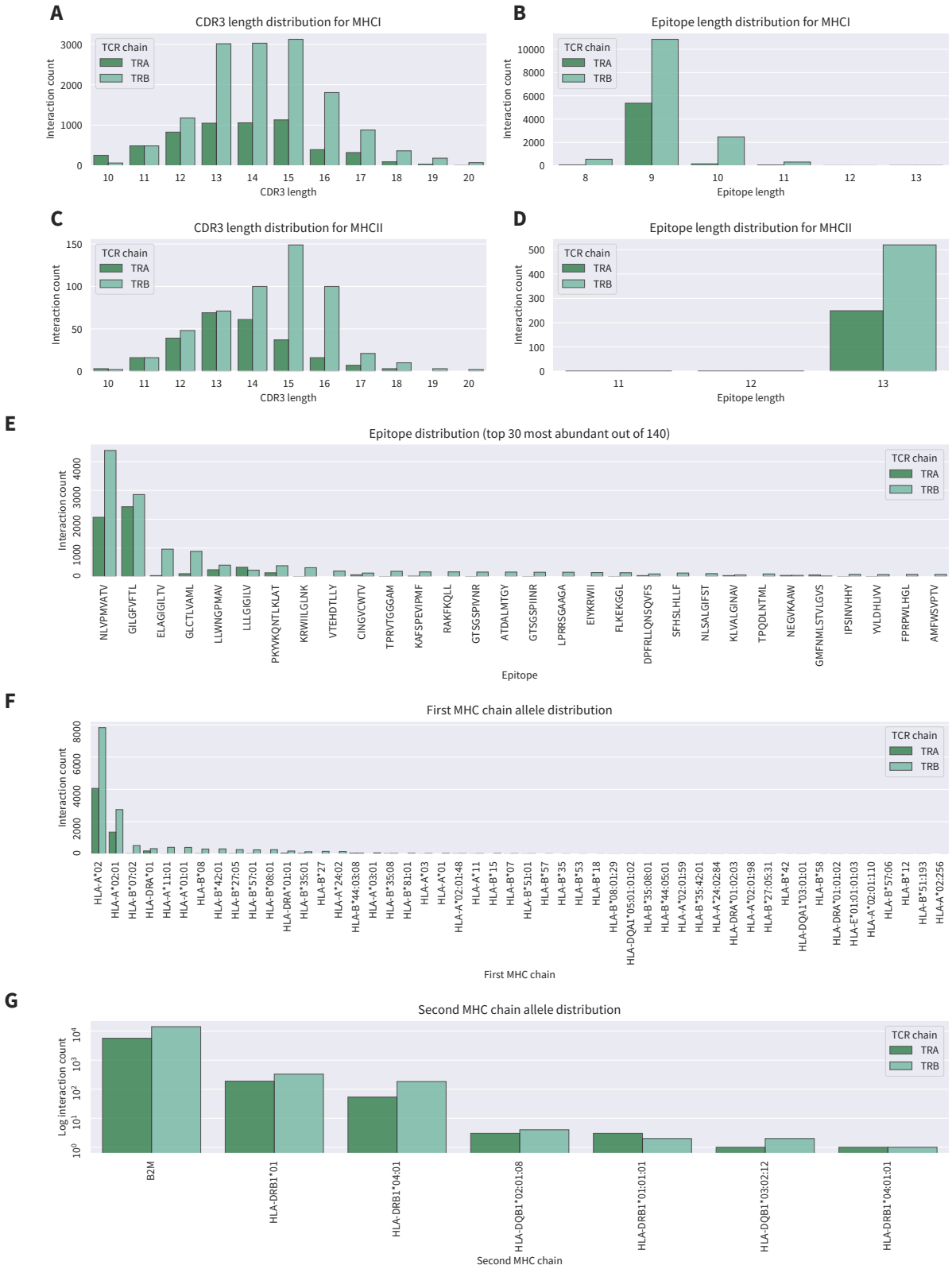

Fig. S1. Overview of contents of the filtered VDJdb data. A, C. Length distribution of TCR CDR3 sequence occurrences for MHCI and MHCII. B, D: Length distribution of epitope sequence occurrences for MHCI and MHCII. E: Top 20 most frequent epitopes. F-G: Abundance of the first and second MHC chain alleles.

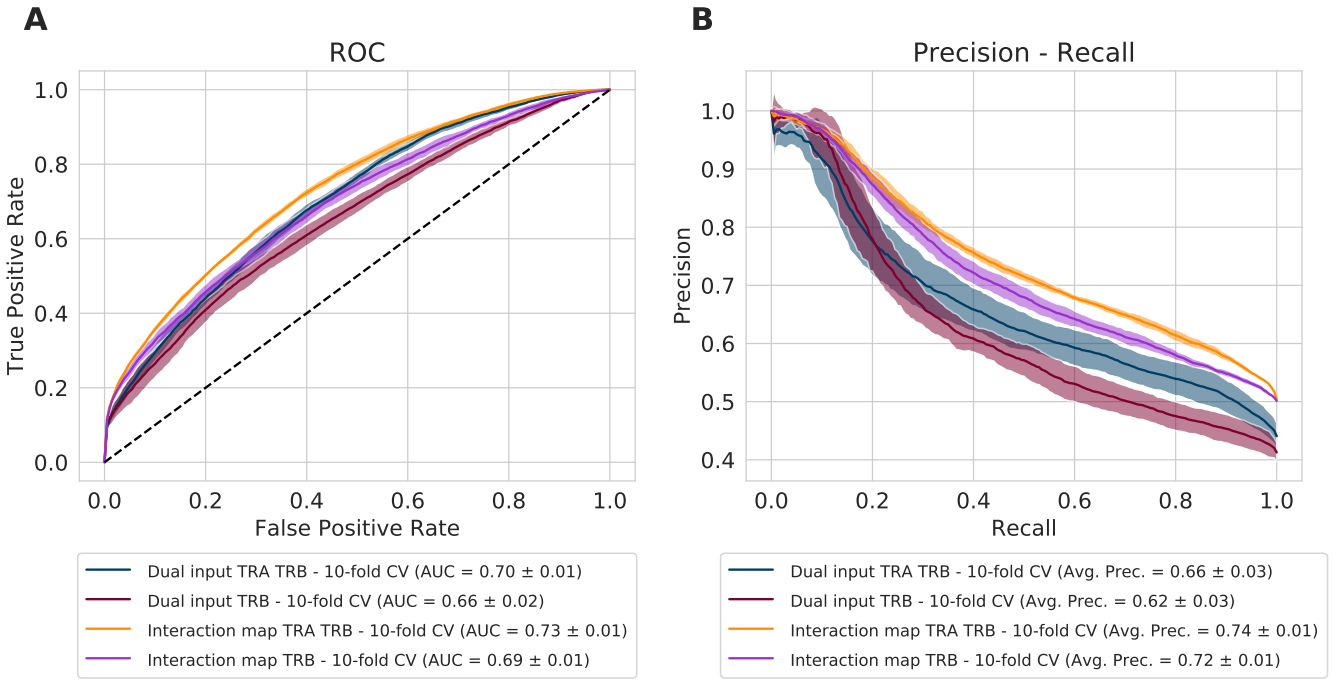

Fig. S2. ROC and PR plot for the interaction map and dual input CNN after 10-fold cross-validation, when trained on the mixed-chain or beta-chain dataset.

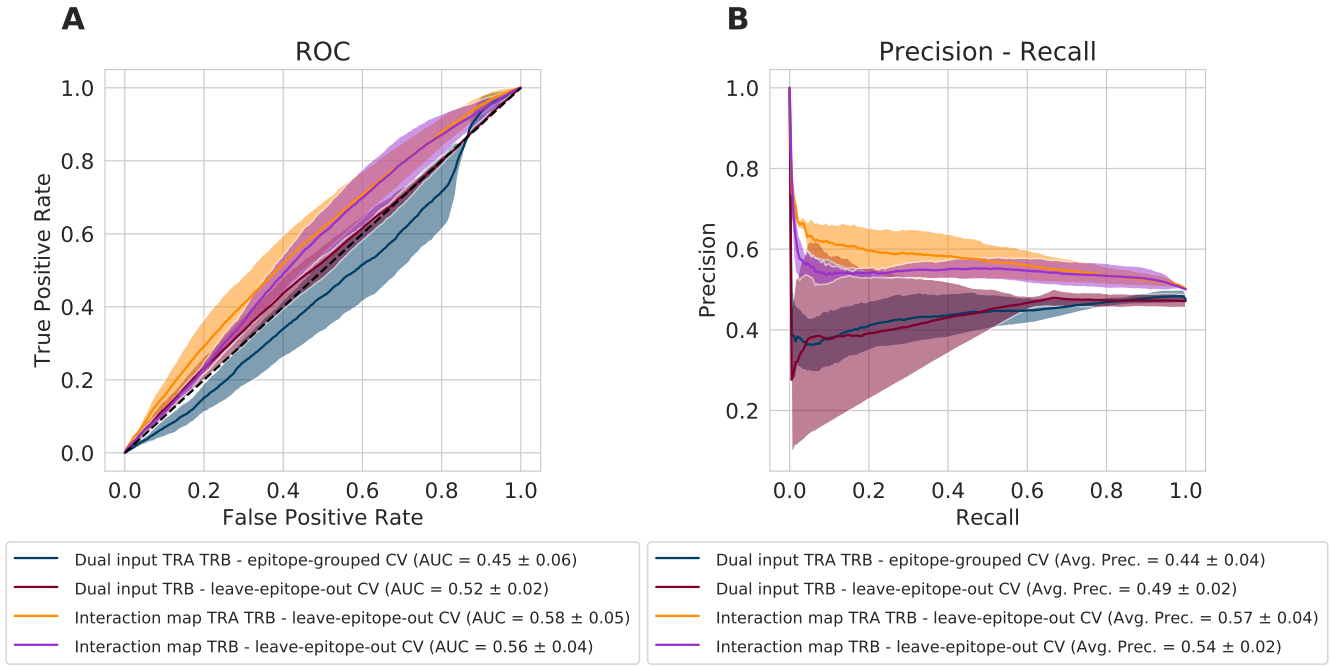

Fig. S3. ROC and PR plot for the interaction map and dual input CNN after epitope-grouped cross-validation, when trained on the mixed-chain or beta-chain dataset.