Universiteit Leiden

Master Computer Science

3D structure-boosted deep learning for the prediction of cancer vaccine candidates

Name:	Heleen Severin
Student ID:	s2991721
Date:	21/02/2025
Specialisation:	Bioinformatics
1st supervisor:	Lu Cao
2nd supervisor:	Li Xue

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

1 Abstract

The Major Histocompatibility Complex (MHC) is a family of proteins responsible for presenting peptides derived from the intracellular degradation of proteins to T-cells. It allows the immune system to recognize non-self peptides from pathogens or transformed cells. Only specific peptide fragments bind to MHC, and the probability of this binding can be described by the binding affinity (BA). Predicting BA is crucial for the development of immunotherapies, such as cancer vaccines. Tools like MHCflurry and NetMHCpan predict BA for peptide-MHC combinations using sequence-based features, but these methods suffer from out-of-distribution (OOD) problems when faced with peptide sequences that differ from the ones seen in training data. This research aims to improve binding predictions by incorporating 3D structural information, as protein shape is critical to its function and ability to interact. peptide-MHC (pMHC) complexes are represented as 3D grids, and 3D convolutions are applied using a 3D CNN architecture. Structures are generated with PANDORA (Marzella et al., 2022), which uses homology modeling and other modeling strategies to produce optimal peptide-MHC conformations. Both positive and negative structures are produced, providing training material containing binding and non-binding interactions. Physico-chemical features are extracted from the structures using DeepRank (Renaud, 2021), and the resulting grid has a 1 Å resolution, with each feature mapped onto a single channel. Experiments with 3D features demonstrate predictive performance competitive with sequence-based methods. Experiments using a train-test split with hierarchically clustered MHCs demonstrate that the network generalizes effectively to unseen MHC alleles, outperforming MHCflurry in the AUC metric. Future research should prioritize reducing data noise and improving on fair benchmarking.

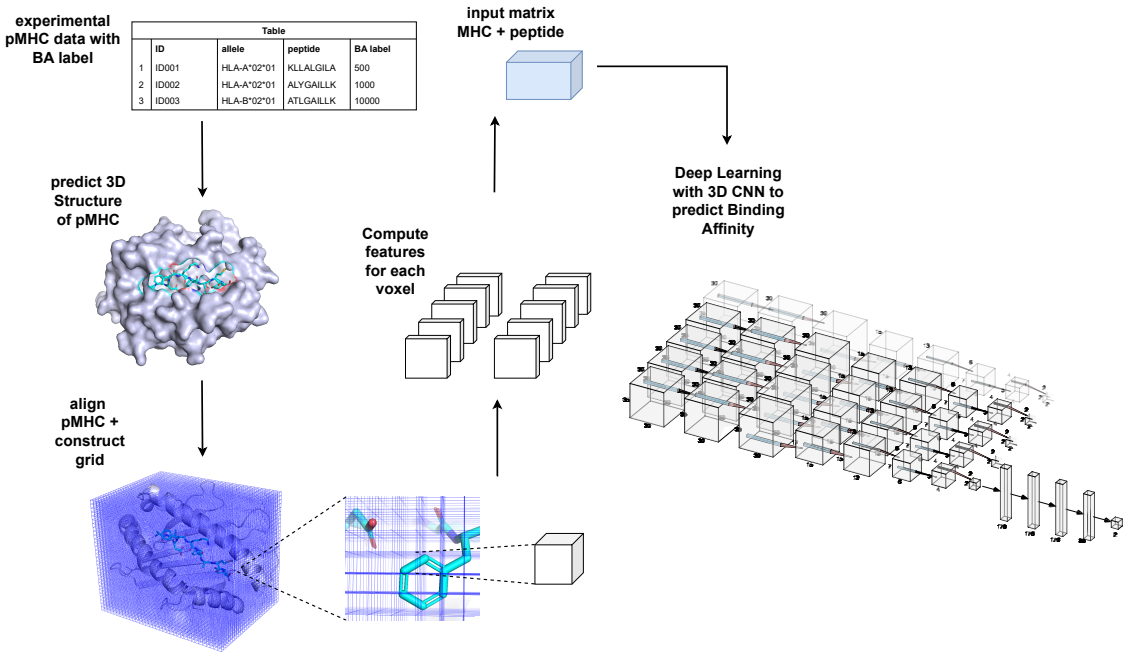


Figure 1: Graphical abstract of experiments workflow: tabular data of peptide-MHC measurements are filtered, 3D structures are rendered by PANDORA, the structures are divided into 3D sections by fitting a grid, physico-chemical features are computed by DeepRank, the resulting 3D matrices serve as an input for a 3D-CNN to train a model for predicting peptide-MHC binding affinities

2 Introduction

2.1 Background

Cancer immunotherapy is a type of treatment that harnesses the body's immune system to identify and combat tumor cells. Currently, a lot of research is being invested in developing several types of these therapies, e.g. monoclonal antibodies, checkpoint inhibitors, cytokines, CAR T-cell therapy, and cancer vaccines. Cancer vaccines do not work preemptively like classical vaccines but have the same principle of administering antigens to combat the disease by mobilizing the immune system [22].

One of the more recent approaches is the use of neo-antigens as a therapy. Abnormal proteins within the cell can originate from viruses and bacteria, but can also be caused by damaged DNA in the cell nucleus that produces faulty proteins. Fragments of the protein strands: peptides, can be recognized by the immune system as non-self. These fragments are called neo-antigens. Tumor cells produce peptides that are never seen before by the immune system and the cells hosting foreign proteins may therefore be selected for elimination. Neo-antigen therapy can be utilized to strengthen the anti-tumor response. Specific neo-antigens, that correspond with the antigens produced by the patient's tumor cells, can be delivered to the patient [30].

The class I Major histocompatibility complex (MHC-I) molecule is a protein present in all nucleated cells of the human body. It is integral for helping the immune system detect foreign proteins in the cell because it binds to (neo-) antigens. MHC is also known as the Human Leukocyte Antigen (HLA) for humans. The MHC-I gene complex consists of the following genes: A, B, C (classical) and E, F, and G (non-classical). MHC-I binds fragments of proteins (peptides) present in the cytosol and travels to the cell surface to present the peptide externally. These peptides are obtained from proteins floating around in the cytosol tagged for destruction. The proteasome will chop off small strings of amino acids, and with the help of TAP it is then transported to the lumen of the ER. The peptide may then be loaded onto a MHC protein. Several post-translational modifications happen in the ER and Golgi complex. After packaging into a vesicle the pMHC (peptide-MHC) multimer can travel to the cell surface to present the antigen on the exterior of the cell (Figure 2) [13]. This collective process is known as 'antigen processing'. This process is selective, meaning that not all peptides will pair up with MHC to be presented on the cell surface. Cytotoxic T-cells (CTL) have specific receptors for antigens: T-cell receptors (TCRs) and play an important role in the adaptive immune system [26]. TCRs will then bind to the pMHC (in this context, the peptides bound to the MHC are referred to as 'antigens'). Once binding is successful, the CTL releases perforins and granzymes, which trigger cytolysis and lead to the death of the host cell. In order to find relevant neo-antigens, high-throughput data from tumor cells are used to screen for target mutations. The antigens (or peptides) should be selected on successful antigen processing) (immunogenicity) and strong binding to relevant MHC-I alleles.

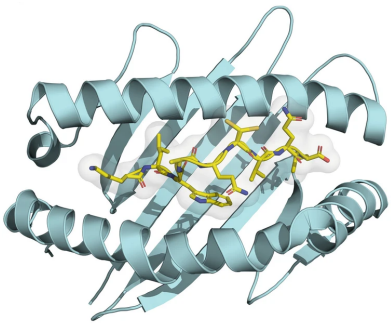


Figure 2: A peptide in MHC binding groove, adapted from Marzella, 2023 [11]

The MHC-I gene complex is highly polymorphic, with the most common and extensively studied genes being HLA-A, HLA-B, and HLA-C. Each person carries a maternal and a paternal allele which are almost always heterozygous. There is an abundance of alleles for each gene, for example, there are 8381 entries for the HLA-A gene in the IPD-IMGT/HLA database [2]. MHC-I can bind to a very large number of peptides, thus pMHCs may be comprised of many combinations of many possible MHCs and peptides. Although a single MHC-I allele is estimated to bind to over tens of thousands of peptides [6], there is a bias in which MHC isoforms can successfully bind to certain peptides. Since there is such an abundance of alleles for MHC-I and a very high number of possible peptides, there

is a need for a computational method that can predict binding affinity between all these possible combinations.

Modeling and predicting the interaction of MHC-I with a peptide is the main objective of this research. MHC-I proteins have a binding groove containing several pockets, of which two are especially relevant. The interaction of the anchoring residues (typically the second and the last residue) of the peptide plays an important role because they have the strongest interaction inside the binding pocket. Ultimately the three-dimensional shape and the chemical and physical properties (eg. polarity, charge) of each residue/atom will play a role in the successful binding of these two molecules.

2.2 Computational approaches

The field of molecular dynamics (MD) employs algorithms that model forces and interactions within the molecules and make predictions about successive molecular configurations. This approach can yield many useful insights into pMHC binding affinity, but is too time-consuming for a high-throughput approach analyzing the vast number of peptide-MHC combinations in the databases (several hundreds of thousands). Most machine learning predictors, however, operate with significantly greater speed, achieving this by relying primarily on sequence information. In essence, these predictors learn the relation between the amino acid distribution of a peptide and every distinct MHC allele. Collecting peptides with positive binding to a specific MHC from experimental data reveals a distinct amino acid distribution. (Figure 3). For a single MHC allele, there will be a certain distribution of amino acids in the peptides that have a positive interaction with the binding groove of an MHC molecule.

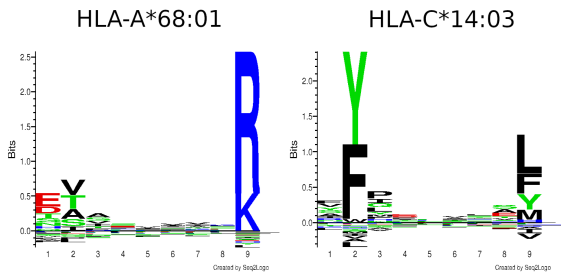


Figure 3: Sequence logos of 9mer peptides with positive binding affinity with respective alleles HLA-A*68:01 and HLA-C*14:13. The x-axis represents the position of the amino acid in the string, the y-axis shows the frequency or probability of a certain amino acid for each position. Position 2 and 9 are the most conserved for 9mers, because they are anchor amino acids.

Sequence-based predictors use a variety of learning methods, most are based on ANN (Artificial Neural Networks). Early methods are allele specific: NetMHC 1 to 3 [8], MHCflurry 1.2.0 [15] which have newer pan-allele methods: NetMHCpan 4xx [7] and MHCflurry 2x [14]. The latter has a combination of immunogenicity network and binding affinity network. MHCNuggets [24] and MATHLA [27] use LSTM networks. Several use ensembles in their methods: MHCflurry 2x, Pickpocket [28] and ForestMHC [4]. Each method uses a different set of features in its learning algorithm, e.g. ForestMHC uses chemical predictions alongside feature information. Other than sequence information, structural information can also be used to predict binding affinity. For example, this is done by ConvNeXt-MHC [29], which encodes the three-dimensional amino acid neighborhoods into a multidimensional matrix.

Learning from sequence information is an effective approach for learning binding affinity, as is shown by the state-of-the-art algorithms. However, this approach suffers from OOD (out-of-distribution) problems when presented with sequence data that is too distant from its learned distributions. Ultimately the 3D shape of the amino acids yields the most important properties to determine interaction on a molecular level. Properties from similar amino acids, such as charge and polarity, should carry over to unfamiliar rearrangements in unseen cases. Therefore the hypothesis is that learning from 3D information will yield a more powerful model for predicting binding affinity and will suffer less from OOD problems.

2.3 3-Dimensional Deep Learning approach

DeepRank [20] is a deep learning framework for making predictions on high-throughput protein-protein complexes. It uses 3D convolutions to gather information from molecule structures. DeepRank is competitive, sometimes outperforming state-of-the-art methods at ranking docking models. The framework can distinguish between crystal artifacts and near-native configurations generated by 3D docking software. The features are derived from physico-chemical properties computed from the binding interface of both peptide and MHC. Contrary to sequence-based predictors, this framework learns based on the 3D interactions between the macromolecules rather than certain amino acid motifs found in the 2-dimensional sequences. DeepRank requires accurate 3D-models of the complexes to calculate the physico-chemical features from the residues and atoms. For this task, accurate pMHC 3D models are needed. To generate these accurate 3D models, PANDORA [10] was selected: a novel modeling pipeline for pMHC complexes developed by the CMBI Structural Bioinformatics group. This algorithm takes a query peptide sequence and allele as input, then identifies an optimal structural template and utilizes a combination of homology modeling and anchor-restrained loop modeling. This approach ensures high-quality models, achieving a 93% success rate within the top 10 generated models, with minimal computational time.

2.4 pMHC datasets

pMHC affinity predictors are typically trained using one or both of the following data types: binding affinity (BA) and Mass Spectrometry (MS). There is a large distinction between these two types of data. BA data are obtained from in vitro experiments, often competitive binding assays. MS data are obtained from in vivo samples (human tissue), yielding a peptidome: a complete set of peptides in complex with MHC. BA data show the likeliness of a random peptide binding to the MHC, but there is no proof that these peptides will go through antigen processing in the cell. Thus BA data only show MHC-peptide combinations that chemically have a high affinity to bind with each other. However, the important caveat with MS data is that these experiments do not reveal which MHC allele was actually bound to the peptides in the peptidome. The nature of the mass spectrometry process makes this determination impossible. To address this limitation, new methods have been developed to 'de-convolute' the original MHC alleles bound to the measured peptides [3]. Additionally, some MS datasets from single-cell lines producing single MHC alleles are available, eliminating the need for de-convolution. Due to their complementary benefits, many modern pMHC predictors, such as MHCflurry and NetMHCpan, utilize a combination of both data types. In essence, MS data provides biologically relevant peptides that have been experimentally verified, while BA data supplies chemically relevant peptides with proven binding characteristics. Biological relevance is crucial because candidate peptides must be processed by the cell to be viable as cancer vaccines. Meanwhile, chemical relevance ensures that the peptides exhibit strong binding to MHC molecules, which is essential for the therapy's efficacy.

3 Methods

3.1 Dataset

The Binding Affinity data for this research was obtained from O'Donnell et al. (2020) [14]. Several filtering steps have been applied to enhance the quality of the labels. 'Qualitative' label and 'inequality' labels are removed to obtain more consistent labels to train the models. After filtering the number of data points is 100,315, with a higher number of negatives than positives (Figure 5b). The total number of alleles is 116, with 29,156 unique peptides. About 72% is represented by gene A and about 27 % by gene B, with only around 1.5% of the data consisting of gene C and others. The measurement data of the BA experiments range from 0 to several million, the distribution without outliers can be seen in Figure 6. Duplicates were kept inside the training data. The collection of binding affinity data has multiple sources and therefore contains several duplicate entries. Figure 4 shows the percentage of duplicates and the number of similar and conflicting labels of those duplicates.

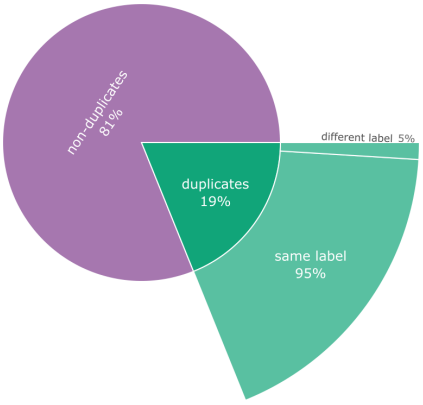
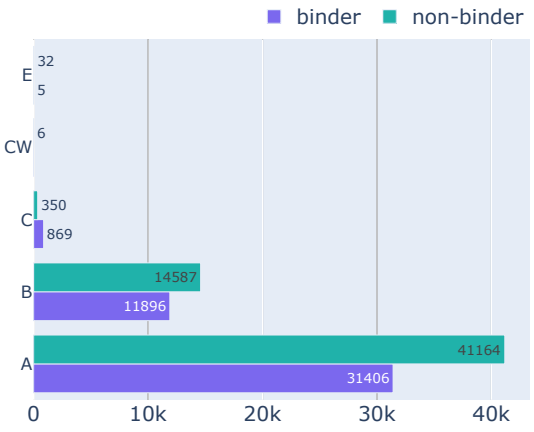
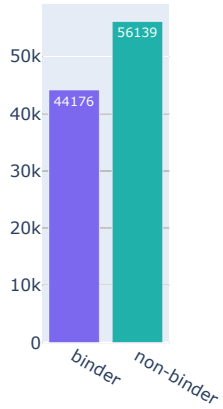


Figure 4: Duplicate peptide-MHC in the dataset: 'same' indicates duplicates meeting the same threshold for binding/non-binding label, while 'different' denotes those that do not.



(a)



(b)

Figure 5: Distributions of binders/non-binders per gene in the filtered BA dataset (a), the overall distribution of binders/non-binders (b)

Mass spectrometry data was not included for training in experiments. However, the integrity of the data was also estimated using these data. The cutoff for BA data to be labeled as positive is an IC50 of < 500nm. Duplicates were identified between the types of data, all MS entries are positives, and most BA entries should have an affinity around the threshold. Figure 16 (supplemental) shows that this holds true but $\sim$30% of the BA data points exceed 500nm and therefore have the negative label, resulting in conflicting labels.

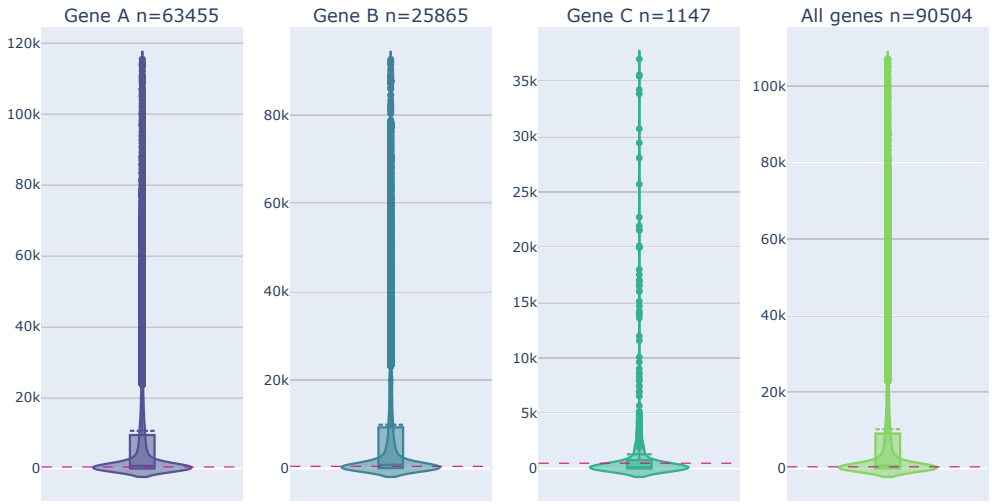


Figure 6: Distributions of the labels/binding affinity measurements in a violin plot overlayed with box plot (the outliers are shown as dots but show a continuous line in some of the plots). Data points above the 99th percentile were removed for this figure. The purple line signifies the cutoff of 500nm. Statistics of the overall genes: mean: 10,3, median: 799, q3: 9190, q1: 55

3.2 Modeling molecular data with DeepRank

DeepRank's approach focuses on encoding the structural information of a 3D molecule by employing a 3D grid. The grid size of the matrix is chosen so that all of the atoms of the binding interface (the residues that are in contact with each other from the two molecules) will fit inside this 3D box. The relation of the number of atoms to voxels is 1:1. Each voxel can hold a number of features calculated for an atom mapped to that voxel. This can also be done for residue features, a residue will span a multiple number of voxels since it is made up of several atoms. For the featurization of molecules, GNNs are often proposed as a method to model the 3D structure, using atoms or residues as nodes and the bonds between those nodes as edges. However, DeepRank addresses the challenge of spatially representing molecules using a grid with limited resolution. Atomic coordinates are mapped onto a 3D grid by identifying the nearest grid point for each coordinate. To improve the resolution, Gaussian mapping is employed: the value of a computed feature is added to grid points in equal directions, and the contribution of each feature follows a Gaussian density. This method eliminates the need for atomic coordinates to align perfectly with grid points. This approach allows feature effects calculated for a specific atom to propagate to its neighbors, the same principle applies to residue features [20].

3.3 Selection of features

The DeepRank package uses atomic and/or residue features to learn the relationship between the properties (e.g. energies) within the molecules and the binding affinity label (Table 1). All features calculated with the DeepRank package [20]. Features are calculated per node: an atom or residue. Certain features are static, like the atom density. Other features are flexible and must be calculated based on the surrounding nodes (e.g. intramolecular charge). The calculated energies are derived from force fields: existing computational models that approximate energies within a network of

atoms.

In addition to the physico-chemical attributes, there is a selection of other features. A PSSM score is also added for all the residues in the MHC-I binding groove. The PSSM scoring matrix was constructed based on a database of all available MHC-I sequences in public databases. In order to obtain a consistent size, a pseudo-sequence (using the most interactive amino acids) was used to represent the MHC. The MHC was then scored against the PSSM scoring matrix. This feature aims to provide information on significant residue changes: an uncommon residue change in a conserved position will have a low probability score. To keep the number of features between the MHC-I and the peptide consistent there is an equivalent feature for the peptide: one-hot-encoded vector for each of the residues. An alternative feature used for the peptide is a skip-gram embedding, specifically trained with peptide epitope data [18]. This describes the frequency of the residues within the peptide (analogous to the PSSM feature). Lastly, there is the anchor feature, a binary indicator that specifies whether a residue is an anchor residue of the peptide. This highlights the two most significant positions in the peptide structure.

The true dimension of the features is doubled since the grid of features always consists of the MHC-I and the peptide chain. This means that features are calculated separately for both the MHC-I and peptide atoms/residues and are concatenated together in one matrix.

Table 1: All features crafted by DeepRank that are computed based on 3D structures of the pMHCs

Feature scale	Feature name	Type	Dimension
Atom level	Atom density	float	5
	Atomic charge	float	1
	Intramolecular electrostatic charge	float	1
	Intramolecular vd Waals energy	float	1
	Desolvation Energy	float	1
Residue level	Buried surface Area (BSA)	float	1
	Residue Contact Density (RCD)	float	7
	Position specific scoring matrix (PSSM)	float	20
	Skip-Gram	float	6
	Anchor	boolean	1

3.4 Data processing

The newly developed 3D-Vac pipeline handles raw input processing, feature crafting, data management, and model training to predict binding affinities using peptide-MHC pairs as input. It processes large tabular datasets containing hundreds of thousands of peptide-MHC pairs in a server environment [17].

3.4.1 Generation of 3D models

PANDORA takes peptide-MHC pairs as input to produce 3D structures of pMHC complexes in PDB format as output. The tool relies on templates of the crystal structure of pMHC complexes. If the MHC protein in the query does not match any of the templates, it uses sequence alignment of the query MHC with all possible template MHC sequences to find the closest match. So any queries where the sequence of the MHC allele is unknown will be excluded from further processing. PANDORA [10] also strongly relies on anchor predictions. Anchors are two amino acids of the peptide that sit in the two main pockets of the MHC binding groove and play a crucial role in the ability of a peptide to bind to the MHC. Anchor predictions can be obtained from computational models trained on similar data. NetMHCpan [21] is used by PANDORA to make such predictions. If such

predictions are not available, the query entry will be excluded. Additionally, PANDORA will use the templates together with the anchor predictions to model the peptide in the best possible configuration (lowest energy) inside the MHC binding groove. This last step is achieved using loop modeling, using the MODELLER [23] software. For downstream processing, only the binding groove domains of the MHC ($\alpha 1$ and $\alpha 2$) and the peptide inside the groove are retained. The final resulting models will have two possible scores associated with them. The scores approximate the free energy, the lower the better the model. Multiple models are produced from different initializations. The best model produced by this algorithm may have configurations that are not possible in biology, for example when residues are clashing. This means that the atoms are too close together to remain in a stable configuration. These models may be produced because the peptide-MHC pair is actually unable to bind physically (non-binder). These cases will have a high energy score associated with them. Obtaining these high-energy models is important to be able to present the self-learning algorithm with negative examples in addition to all the positive examples.

3.4.2 Feature generation

The focus of the analysis of the pMHC is on the binding interface. Therefore, DeepRank takes a distance cutoff of 15 Å around the interacting atom between the two molecules. All the pMHC models are aligned with GradPose [19] to one reference pMHC to obtain a consistent orientation between all examples with all equivalent amino acids on the same coordinates. The grid size was optimized to the shape of the binding interface and has the following dimension: $35 \cdot 30 \cdot 30$ $\mathring{A}^3$. The corresponding input matrix holding the features has the equivalent amount of voxels. The atomic and residue features are computed for all grid points for both the peptide and the MHC. These features and their indices are saved in HDF5 [25] format by the DeepRank package to create the input matrix on demand when training is initialized. All features are min-max normalized and standardized before training. All labels are binary, where a cutoff of < 500nm is used for positives.

3.4.3 Data partitioning

The partitioning of the peptide-MHC pairs for the train, validation, and test set was accomplished using three methods: random shuffling, clustering on peptide similarity and lastly clustering on MHC similarity. The peptide similarity clustering was done with GibbsCluster 2.0 ([1]), it is an unsupervised clustering method aiming to find meaningful groups by identifying motifs in the peptide sequences. The parameters used were peptide length=15, and number of clusters=10. The MHC-I similarity clustering was performed by scoring all the individual genes against a PAM30 matrix to obtain evolutionary distances. The dendrogram resulting from this was used to find the most distant clusters of alleles to be used in the test set, each around 10% of the total data points corresponding to genes A, B and C. Five different clusters were obtained.

3.5 Experiments

Three categories of experiments were performed to test the model's generalization power. First testing the model's ability to generalize to alleles from an unseen gene, specifically gene C. This task involved training the network on a set of common MHC-I alleles and evaluating uncommon MHC-I alleles (along with their respective peptide pairs). The training/validation dataset included the full dataset, excluding all alleles from gene C, while the test set consisted entirely of alleles from gene C. The split resulted in 72% positives and 28% negatives (see Figure 14) All features were included except for the skip-gram feature and atom densities.

Second, the effect of subsets of features used in the network was tested. Theoretically, the model should be able to represent protein interactions on a molecular (i.e. residue) or atom level. These two levels are measured against a combined model using both atom and residue features. The atom model was implemented with the following features: desolvation energy, atomic charge, intramolecular charge, vd Waals and atomic densities. The Residue model: BSA, RCD, Skip-Gram and

anchor. The combined model: desolvation energy, RCD, Skip-Gram, BSA, atomic charge, intramolecular charge and vd Waals (section 3.3). The test set consisted of 59% negatives and 41% positives (see Figure 14).

Finally, the results are compared to the state-of-the-art binding affinity prediction algorithm. MHCflurry 2.0 [14] is a deep learning prediction tool that utilizes only the sequence information of the peptides and the MHC. MHCflurry 2.0 is compared against DeepRank CNN with randomly shuffled data and clustered data. The CNN used the following features: desolvation energy, RCD, Skip-Gram, BSA, atomic charge, intramolecular charge and vd Waals and anchor. This experiment was performed 5-fold with the five different clusterings. These experiments had around 40% negative and 60% positive distribution (see Figure 15).

3.6 Deep learning

The HLA-C left-out experiment was performed with architecture 1, found in Table 2. In summary: The first layer is batch normalization for 3D, this is followed by a 3D convolutional layer + ReLu where the n features are halved to function as a projection layer [12], then 3 layers of 3D convolution (kernel size 2) + max 3D pooling + ReLu. Then the layers are flattened. In the feed-forward part, batch normalization is applied, a linear layer follows that maps the flattened array to 1000 + ReLu + 0.5 dropout, then linear layer + ReLu + 0.5 dropout, keeping the dimension of 1000. Finally, the output layer is a linear layer mapping to an output of 2. The batch size was 128, the optimizer 'sgd', the learning rate 1e-3, a cross-entropy loss function was used and the training was done with a maximum of 40 epochs.

The feature set experiment was performed with an architecture similar to architecture 1 with some modifications: the kernel size was set to 3 (apart from the first layer), ELU was used instead of ReLu, the fully connected layers halved the dimension for each layer from the flattened dimension. Other parameters were the same except for a maximum of 25 epochs.

The SOTA experiment CNN was trained with the following architecture (see Table 6): the first layer is batch normalization, followed by a 3D convolutional layer with ReLU, effectively a projection layer where the number of features is halved. Then convolutional layers + ReLu with kernel size 3, followed by batch normalization. Then two convolution blocks with ReLu and batch normalization where the features space is doubled again. Max pooling then reduces the space of the grid to about half the size. The convolutional block is followed by a feed-forward network where the grid is flattened and reduced to 128, three linear layers follow where ReLu and dropout are applied. Finally, softmax is applied to the output. The batch size was 128, the optimizer 'Adam' with learning rate 1e-3 and cross-entropy loss function, and the training lasted 15 epochs. The training details for the MLP and MHCflurry can be found in the supplemental materials of Marzella et al. (2024) [9].

4 Results

4.1 Leaving out allele group

The full dataset was used as the training/validation set in this experiment, excluding all alleles from gene C. Figure 9a shows that the loss curve did not flatten completely after a training time of 40 epochs, but the loss was still high at ∼0.5. Figure 7 shows that after the learning phase, the model was unable to generalize to alleles of HLA-C to make reliable predictions. It shows that at initialization, the model scores the expected random ROC-AUC score of ∼0.5 and after 40 iterations scores worse than random with an AUC of only 0.481.

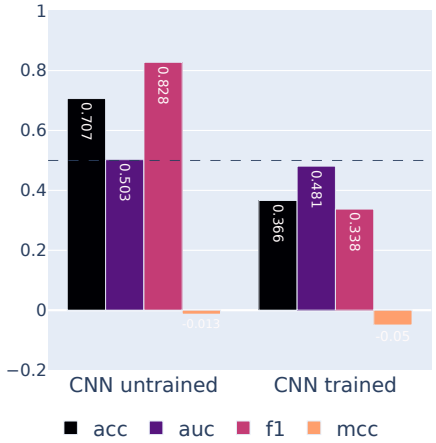


Figure 7: Performance of DeepRank CNN on test set containing only HLA-C alleles, not seen in train set

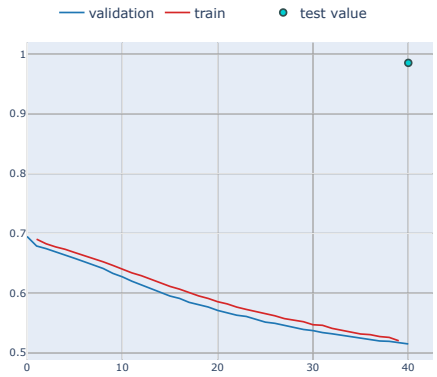
4.2 Feature set experiments

To know which features contribute to the performance or might cause the model to overfit, different feature sets were tested. The results in Figure 8 show that the model trained with just atom features slightly outperforms the model using combined features, which again outperforms the model trained with residue features. The especially low MCC score for the residue model can be explained by the very low TPR (0.161) of the model. Combined with a high TNR (0.90), this indicates that the model was skewed towards predicting negative labels in the majority of the cases. This is also true to a certain degree for the 'combined features' and 'atom features' model where the TPR is almost twofold compared to the TNR and more than twofold in the case of the 'combined features' model. The bias of the models is also seen in the density plots (Figure 10) The data points from the validation sets of the experiments did not resemble the test set enough to indicate a good stopping point for learning (Figure 9), thus the number of epochs was likely not optimized for these experiments. The Residue + Atom model (Figure 9b) suggests overfitting because the training loss is slightly lower than the validation loss and keeps diverging with increasing epochs.

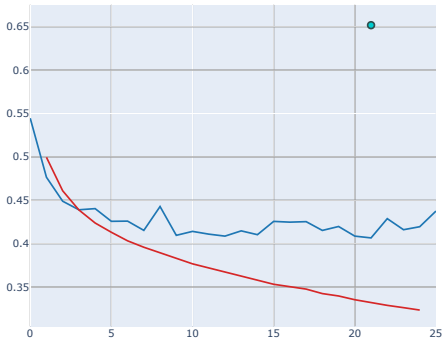
Figure 8: Performance of DeepRank CNN on varying feature sets (section 3.5), with random selection of peptide-MHC pairs for train-test

4.3 Clustering experiments against state-of-the-art

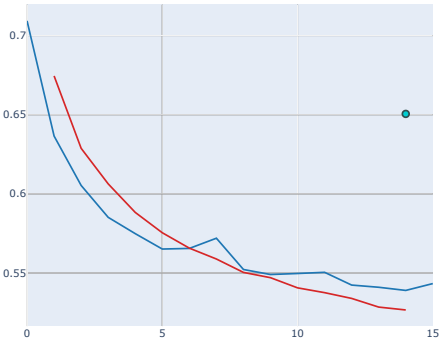
A simple Deep Learning algorithm (MLP) and a state-of-the-art algorithm (MHCflurry 2.0) were tested against the DeepRank CNN with three sets of clusters (Figure 13). All three methods (MLP, CNN, MHCflurry) performed best on random clustering and showed the lowest performance on MHC clustering, based on AUC. The simplest architecture (the MLP) scored the highest with both imbalanced metrics (ACC and AUC) and balanced metrics (MCC and F1). The AUC drop between clustered and shuffled is 0.149 (CNN), 0.334 (MLP), and 0.163 (MHCflurry).



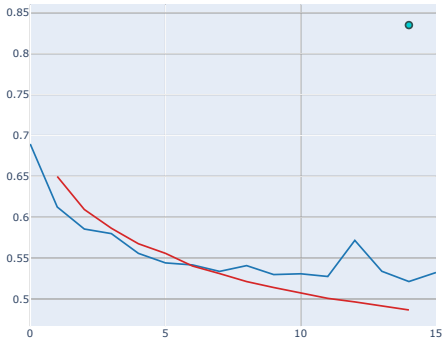
(a) Losses per epoch for HLA-C left out model



(b) Losses per epoch for Residue + Atom feature model



(c) Losses per epoch for Atom features model



(d) Losses per epoch for Residue features model

Figure 9: Losses during training of four different experiments with DeepRank CNN: HLA-C left out and three features sets experiment: atom + residue, atom features, residue features

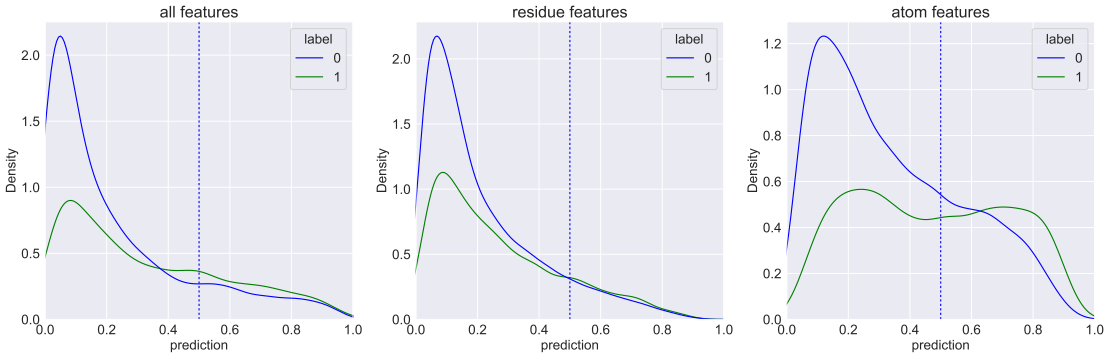


Figure 10: Density plot of test set probabilities for three DeepRank CNN models in feature experiments. A perfect model has two separate curves on each side of the 0.5 vertical line, positives $\geqslant 0.5$ and negatives < 0.5. The atom model has the best separation of the three experiments, though the density for the positive label is almost evenly distributed across the probability space

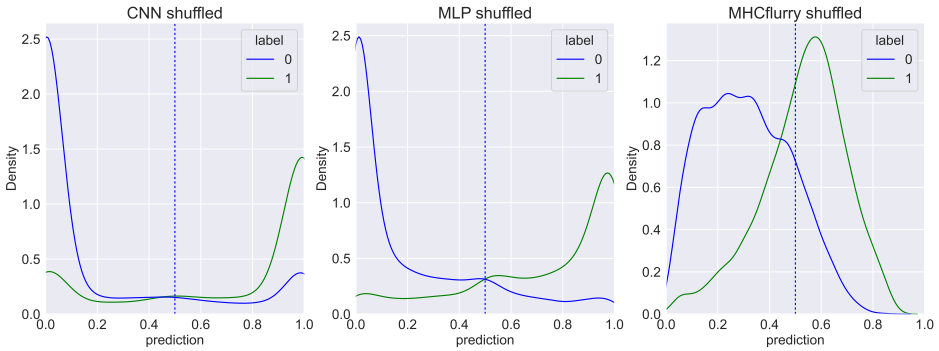


Figure 11: Density plot of the probabilities on the test set from three different models, DeepRank CNN, MLP and state-of-art (MHCflurry) with shuffled train-test. A perfect model has two separate curves on each side of the 0.5 vertical line, positives $\geqslant 0.5$ and negatives < 0.5.

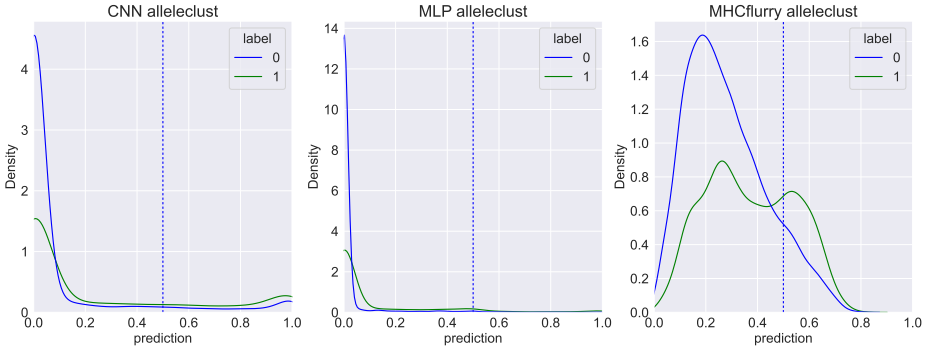
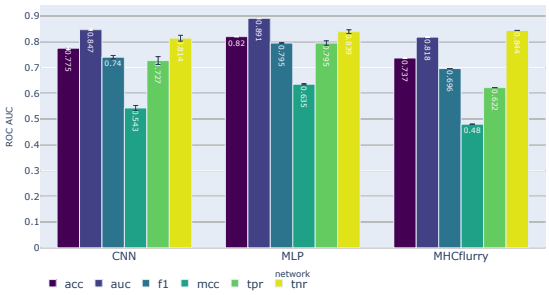
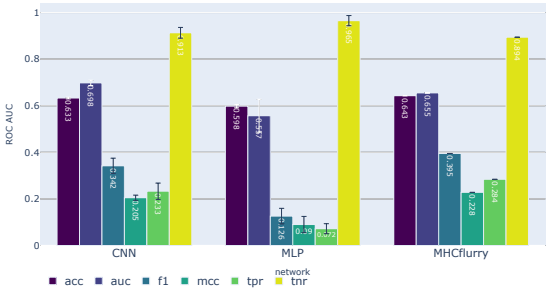


Figure 12: Density plot of the probabilities on the test set from three different models, DeepRank CNN, MLP and state-of-art (MHCflurry) with allele clustering to separate train-test. A perfect model has two separate curves on each side of the 0.5 vertical line, positives $\geqslant 0.5$ and negatives < 0.5.

The density plots for the state-of-the-art experiments (see Figure 11 and 12) show a notably different prediction distribution for the MLP, with shuffling there is clear separation, while clustering shows poor separation. The CNN shows a clear decrease in successfully predicting positives with clustering compared to the shuffled experiments. The same is true for MHCflurry. Figure 13 also shows that the true positive rate (TPR) drops significantly for all models between clusterings, while the true negative rate (TNR) stays high and even increases slightly for all models. The DeepRank CNN shows the highest ROC AUC score between the three models for MHC clustering. However, MHCflurry has a higher F1 and MCC score, indicating a better balance between type 1 and type 2 errors.



(a) Random clustering (shuffled)



(b) Clustering by MHC similarity

Figure 13: Performance on the test set from three different models, DeepRank CNN, MLP, and state-of-art (MHCflurry) with two modes of clustering for the training and test set: random (shuffled), clustered by MHC similarity

5 Discussion

5.1 Benchmarking and ranking

To this date, there is no universal benchmark for pMHC affinity prediction methods. New algorithms often benchmark their new method against other state-of-the-art methods using a test dataset according to self-motivated criteria. Sometimes test data is selected on the criteria that it is 'unseen' by the trained algorithms, however, unseen alleles are rarely a criterion and the similarity of the peptide amino acid distribution is often not considered. Additionally, each research publication chooses one or two metrics to compare and rank the tested algorithms, which makes comparing benchmarks across publications even more difficult. Popular metrics are among others: AUCROC (area under the curve of Receiving Operating Characteristic), AUCPRC (area under the curve of Precision and Recall), Accuracy, and top n% ranking (number of positives found for data points with top n probability). The widely used ROCAUC does not take the label distribution of the test set into account, Mathews correlation coefficient (MCC) is the most suited metric when false positives and false negatives are of equal importance [5]. A variety of metrics should be reported to increase comparability. The most popular methods (MHCflurry, NetMHCpan) use data from a source separate from their training data for final validation (commonly referred to as 'test set') to ensure a fair evaluation of their predictive performance. However, there is no focus on the overlap of alleles and peptides as a criterion for selecting these data. In O'Donnell (2020) [14] there is a 100% overlap of MHC alleles and a large majority of overlap in peptides in both sets.

5.2 Data limitations

Training data is abundant for certain alleles, especially HLA-A:02*01, but the frequency in BA datasets does not always represent the frequency in the human genome. Alleles of gene C of MHC are very rare in general in pMHC datasets, while it could provide important insights for pMHC binding affinity research. Computational models should be able to perform well on rare alleles to be relevant for advanced research in personalized medicine. Because of the discrepancy between allele frequencies in datasets and the frequency in biology, these models will not perform in real-world cases where rare alleles could be present. Another crucial link for immunogenicity is TCR compatibility. The CD8+ T-cell must bind to the pMHC complex to illicit a tumor-specific CTL response. The lack of experimental data for TCR:pMHC bindings explains the current scarcity of predictors taking TCR binding into account. Nevertheless, MHC-peptide interaction after antigen processing is thought to be the single most selective factor for immune response.

5.3 Exclusion of mass spectrometry data

The increasing availability of experimental MHC epitope data facilitates the development of algorithms with improved predictive power, driven by the greater volume and diversity of training data. Some argue that models trained solely on chemically validated BA labels are becoming less relevant, MS data provides biologically validated evidence of MHC-peptide binding. On the other hand, a case can be made for developing models that focus on accurately representing the chemical interactions between MHC and peptides, independent of antigen processing properties. For vaccine design applications, such models should be integrated with an antigen processing model to ensure that predicted peptides possess immunogenic properties. An undeniable drawback of excluding MS data is the loss of training data volume to further improve the robustness of the model. However, the bias that MS data introduces to the peptide landscape could be offset by including enough chemical (BA) examples in the training data.

5.4 Data quality Limitations

BA labels are not consistent between duplicates and the control (MS data) as shown by the analysis in 3.1. The binding affinity cutoff also does not correspond to the positive data points that MS datasets provide, where almost 30% exceed the cutoff of 500nm and are falsely labeled as non-binders. To determine a better cutoff, more research needs to be done, also accounting for biological negatives, not just positives. The binding affinity cutoff has been proven specific to the allele: a successful binding in vivo corresponds to a different binding affinity measurement [16]. Inconsistencies between duplicate BA data points also suggest that the labels are unreliable to a certain degree. From all the duplicate data points, 5% had conflicting labels. The lacking quality of the labels is also likely causing troubles for any algorithm to distinguish the binders from the non-binders. Likely the quality of the non-binders is better than the binders. This is because the range of the IC measurement beyond the cutoff of 500nm is very high: it goes up to hundreds of thousands in certain cases (see Figure 16). This theory is also supported by the model's tendency to identify negatives with confidence but not the positives (see Figure 10 The data enrichment step could also possibly introduce some quality issues: high-energy 3D models of pMHC may be produced by PANDORA while the complex should have low energies in biology. This will limit the Deep Learning model from learning the correct relationships since these complexes have a positive BA associated labels, while the energy of the calculated features is high. Because it requires manual inspection, the exact number of low-quality 3D models in the dataset is not known. Additionally, the generated 3D structures (in PDB format) lacked properly assigned hydrogen atoms, leading to inaccuracies in energy calculations.

5.5 Experiment flaws

Multiple iterations of training a model starting from a random state are important to determine the stability or the variance in performance. Due to extensive training times and a limited budget, not all experiments had replicates, only the state-of-the-art comparison. Because of this limitation, the metrics may slightly over- or underestimate the performance. Furthermore, the feature set experiment suggests that atom features yield models with the best predictive power. However, the skip-gram and anchor feature should have been included in the 'combination feature' model. Due to the omission of these features, it is unknown whether the inclusion of those features was causing the drop in performance in the residue model, or if it was due to the lack of atom features.

To accurately determine feature importance and identify the optimal feature set, an ablation test should be conducted. The ablation test removes any assumptions about how the model should theoretically be constructed (e.g. on atom or residue level) and finds the optimal set based on empirical results. Ideally an Automated Machine Learning (AML) method is used to find the ideal set without running all the permutations. However, the number of parameters of the experimental models was too large to justify the cost of running an AML application with the resources available.

5.6 Future research

The research and experiments conducted in this study are closely related to the work of Marzella et al. (2024). In this new experimental work, the CNN is tested against a GNN (using similar physicochemical features to the CNN in the Results 4) and an eGNN (using only amino acid types and distances as features). The (e)GNN represents the atoms as nodes, and the edges are equivalent to the relative distances between the atoms. In allele cluster experiments (similar to this experiment in this report) GNN outperforms the CNN and again the eGNN outperforms GNN. From these results can be hypothesized that a GNN is better suited to represent a molecule and its interactions in 3D space compared to a constrained grid that was used for the CNN. The resolution (1 Å) of the grid is simply not sufficient to approximate the true geometry of the molecules like a GNN can do inherently. The Gaussian mapping is supposed to alleviate the restriction of the mapping to a constrained grid, but this representation might be too distant from a high-resolution 3D model for the Deep Learning model to learn good correlations. A higher resolution might boost performance

but will be exponentially slower and can never match the efficiency of representing and processing power of a GNN. Another point of discussion is how the features of the MHC chain and peptide chain are combined as a single data point. The features of both chains are represented on a separate matrix. This method was chosen to make a clear distinction between MHC and ligand grid points. Because the matrices of the chains are concatenated together the model needs to learn the relationship between the interactions on equivalent coordinates of the separate matrices. This introduces another challenge and potentially harms learning efficiency. Additionally, the rotational invariance of the GNN makes the direction of interaction between atoms or residues with respect to each other irrelevant. This presumably, gives the GNN a big advantage in learning quality and efficiency. The eGNN outperforming the GNN suggests that, rather than explicitly providing physico-chemical properties, geometry is the ultimate most important feature to represent the molecules, which is something that the CNN inherently cannot represent.

6 Conclusion

The OOD problem is known across different fields of science where data-driven modeling is used. A way to improve modeling for protein-protein interaction is to obtain more robust features from the interaction of these molecules. Rather than just 2D sequence information, DeepRank incorporates 3D structural information of the molecules. Different feature subsets were tested to find the influence on performance. The findings from the model with atomic-only features showed that performance was actually superior without a sequence embedding in its feature set. Additionally, the DeepRank CNN was tested against state-of-art in different clustering scenarios. The performance of the CNN models shows that this structural method is competitive with state-of-the-art in both shuffled learning scenarios as well as unseen groups of MHC alleles. Based on the commonly used AUC score, it even outperforms the state-of-the-art in the MHC-cluster experiment. However, the state-of-the-art network still performs better in recall, a relevant metric for this task. Subsequent research has already shown that structure (or geometry) is an excellent modeling strategy for this task. Future research should also focus on reducing data noise for reliable modeling, as inconsistent labeling hinders learning.

7 Supplemental

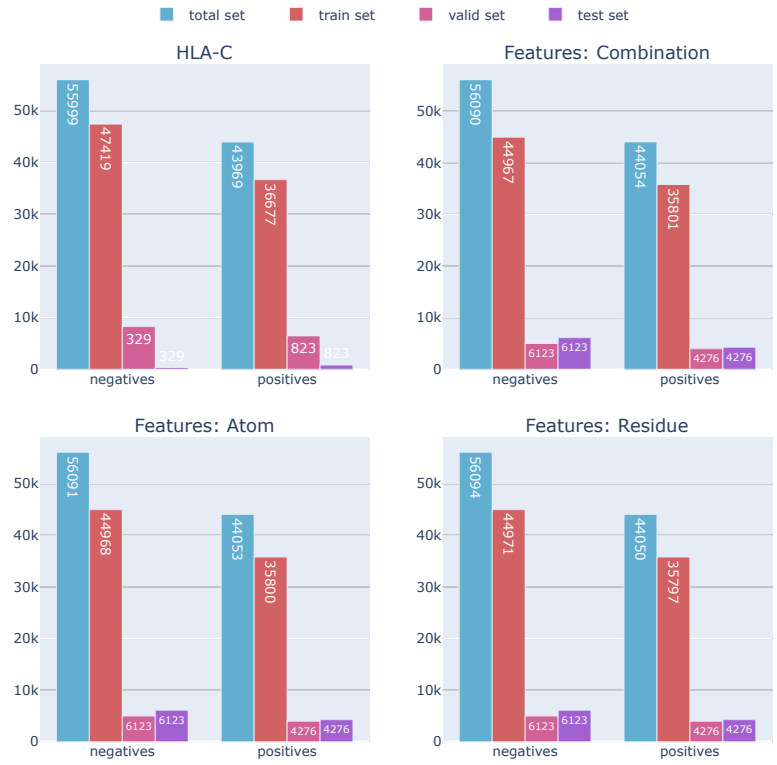


Figure 14: Distributions of positives and negatives for the features subset experiments with the DeepRank CNN

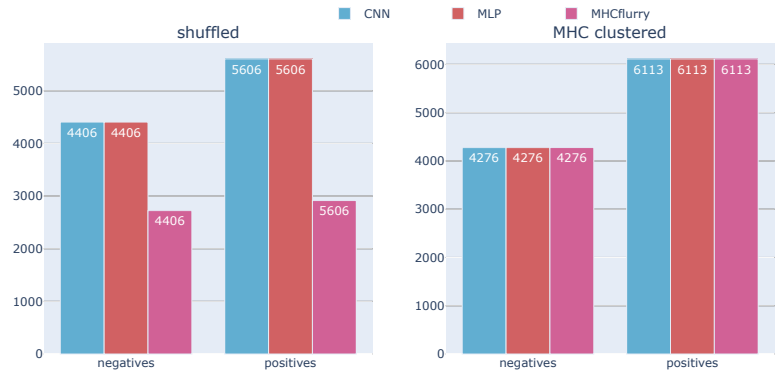


Figure 15: Distributions of positives and negatives in the test set for the state-of-the-art clusters experiments

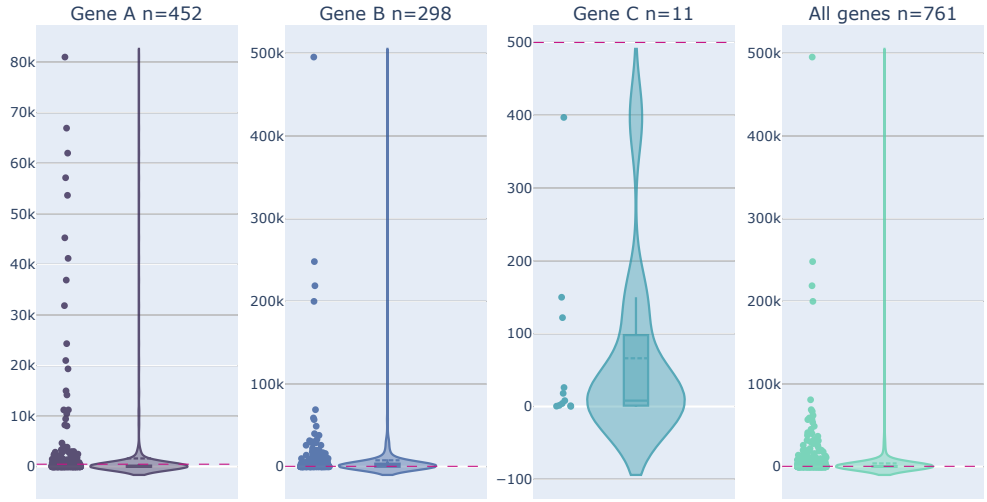


Figure 16: Overlapping BA experiments with MS experiments. All the overlapping BA should have a positive label since MS is the biologically proven label. The threshold of 500nm (purple dotted line) captures the majority of the BA data points looking at the overall genes (median=67, q1=7, q3=792.5) but still a lot of positives are lost due to the cutoff: about 30 %.

Table 2: **Architecture 1 (HLA-C)**:
Total params: 2,231,514
Trainable params: 2,231,514
Non-trainable params: 0
Input size (MB): 10.57
Forward/backward pass size (MB): 56.92
Params size (MB): 8.51
Estimated Total Size (MB): 76.01

Layer (type)	Output Shape	Param
BatchNorm3d-1	[-1, 88, 35, 30, 30]	176
Conv3d-2	[-1, 44, 35, 30, 30]	3,916
ReLU-3	[-1, 44, 35, 30, 30]	0
Conv3d-4	[-1, 44, 35, 30, 30]	1,980
MaxPool3d-5	[-1, 44, 17, 15, 15]	0
ReLU-6	[-1, 44, 17, 15, 15]	0
Conv3d-7	[-1, 44, 16, 14, 14]	15,532
MaxPool3d-8	[-1, 44, 8, 7, 7]	0
ReLU-9	[-1, 44, 8, 7, 7]	0
Conv3d-10	[-1, 44, 7, 6, 6]	15,532
MaxPool3d-11	[-1, 44, 3, 3, 3]	0
ReLU-12	[-1, 44, 3, 3, 3]	0
Flatten-13	[-1, 1188]	0
BatchNorm1d-14	[-1, 1188]	2,376
Linear-15	[-1, 1000]	1,189,000
ReLU-16	[-1, 1000]	0
Dropout-17	[-1, 1000]	0
Linear-18	[-1, 1000]	1,001,000
ReLU-19	[-1, 1000]	0
Dropout-20	[-1, 1000]	0
Linear-21	[-1, 2]	2,002

Table 3: **Architecture 2 (Atom & Residue features)**:
Total params: 75,022
Trainable params: 75,022
Non-trainable params: 0
Input size (MB): 5.29
Forward/backward pass size (MB): 28.25
Params size (MB): 0.29
Estimated Total Size (MB): 33.83

Layer (type)	Output Shape	Param
BatchNorm3d-1	[-1, 44, 35, 30, 30]	88
Conv3d-2	[-1, 22, 35, 30, 30]	990
ELU-3	[-1, 22, 35, 30, 30]	0
Conv3d-4	[-1, 22, 35, 30, 30]	506
MaxPool3d-5	[-1, 22, 17, 15, 15]	0
ELU-6	[-1, 22, 17, 15, 15]	0
Conv3d-7	[-1, 22, 15, 13, 13]	13,090
MaxPool3d-8	[-1, 22, 7, 6, 6]	0
ELU-9	[-1, 22, 7, 6, 6]	0
Conv3d-10	[-1, 22, 5, 4, 4]	13,090
MaxPool3d-11	[-1, 22, 2, 2, 2]	0
ELU-12	[-1, 22, 2, 2, 2]	0
Flatten-13	[-1, 176]	0
BatchNorm1d-14	[-1, 176]	352
Linear-15	[-1, 176]	31,152
ELU-16	[-1, 176]	0
Dropout-17	[-1, 176]	0
Linear-18	[-1, 88]	15,576
ELU-19	[-1, 88]	0
Dropout-20	[-1, 88]	0
Linear-21	[-1, 2]	178

Table 4: **Architecture 3 (Atom features)**:
Total params: 12,791
Trainable params: 12,791
Non-trainable params: 0
Input size (MB): 2.16
Forward/backward pass size (MB): 11.56
Params size (MB): 0.05
Estimated Total Size (MB): 13.77

Layer (type)	Output Shape	Param
BatchNorm3d-1	[-1, 18, 35, 30, 30]	36
Conv3d-2	[-1, 9, 35, 30, 30]	171
ELU-3	[-1, 9, 35, 30, 30]	0
Conv3d-4	[-1, 9, 35, 30, 30]	90
MaxPool3d-5	[-1, 9, 17, 15, 15]	0
ELU-6	[-1, 9, 17, 15, 15]	0
Conv3d-7	[-1, 9, 15, 13, 13]	2,196
MaxPool3d-8	[-1, 9, 7, 6, 6]	0
ELU-9	[-1, 9, 7, 6, 6]	0
Conv3d-10	[-1, 9, 5, 4, 4]	2,196
MaxPool3d-11	[-1, 9, 2, 2, 2]	0
ELU-12	[-1, 9, 2, 2, 2]	0
Flatten-13	[-1, 72]	0
BatchNorm1d-14	[-1, 72]	144
Linear-15	[-1, 72]	5,256
ELU-16	[-1, 72]	0
Dropout-17	[-1, 72]	0
Linear-18	[-1, 36]	2,628
ELU-19	[-1, 36]	0
Dropout-20	[-1, 36]	0
Linear-21	[-1, 2]	74

Table 5: **Architecture 4 (Residue features)**:
Total params: 35,087
Trainable params: 35,087
Non-trainable params: 0
Input size (MB): 3.60
Forward/backward pass size (MB): 19.26
Params size (MB): 0.13
Estimated Total Size (MB): 23.00

Layer (type)	Output Shape	Param
BatchNorm3d-1	[-1, 30, 35, 30, 30]	60
Conv3d-2	[-1, 15, 35, 30, 30]	465
ELU-3	[-1, 15, 35, 30, 30]	0
Conv3d-4	[-1, 15, 35, 30, 30]	240
MaxPool3d-5	[-1, 15, 17, 15, 15]	0
ELU-6	[-1, 15, 17, 15, 15]	0
Conv3d-7	[-1, 15, 15, 13, 13]	6,090
MaxPool3d-8	[-1, 15, 7, 6, 6]	0
ELU-9	[-1, 15, 7, 6, 6]	0
Conv3d-10	[-1, 15, 5, 4, 4]	6,090
MaxPool3d-11	[-1, 15, 2, 2, 2]	0
ELU-12	[-1, 15, 2, 2, 2]	0
Flatten-13	[-1, 120]	0
BatchNorm1d-14	[-1, 120]	240
Linear-15	[-1, 120]	14,520
ELU-16	[-1, 120]	0
Dropout-17	[-1, 120]	0
Linear-18	[-1, 60]	7,260
ELU-19	[-1, 60]	0
Dropout-20	[-1, 60]	0
Linear-21	[-1, 2]	122

Table 6: **Architecture 5 (CNN SOTA)**:
Total params: 1,629,026
Trainable params: 1,629,026
Non-trainable params: 0
Input size (MB): 2.88
Forward/backward pass size (MB): 35.55
Params size (MB): 6.21
Estimated Total Size (MB): 44.64

Layer (type)	Output Shape	Param
BatchNorm3d-1	[-1, 24, 35, 30, 30]	48
Conv3d-2	[-1, 12, 35, 30, 30]	300
BatchNorm3d-3	[-1, 12, 35, 30, 30]	24
ReLU-4	[-1, 12, 35, 30, 30]	0
Conv3d-5	[-1, 12, 35, 30, 30]	156
BatchNorm3d-6	[-1, 12, 35, 30, 30]	24
ReLU-7	[-1, 12, 35, 30, 30]	0
Conv3d-8	[-1, 24, 33, 28, 28]	7,800
BatchNorm3d-9	[-1, 24, 33, 28, 28]	48
MaxPool3d-10	[-1, 24, 16, 14, 14]	0
ReLU-11	[-1, 24, 16, 14, 14]	0
Conv3d-12	[-1, 48, 14, 12, 12]	31,152
BatchNorm3d-13	[-1, 48, 14, 12, 12]	96
MaxPool3d-14	[-1, 48, 7, 6, 6]	0
ReLU-15	[-1, 48, 7, 6, 6]	0
Flatten-16	[-1, 12096]	0
BatchNorm1d-17	[-1, 12096]	24,192
Linear-18	[-1, 128]	1,548,416
ReLU-19	[-1, 128]	0
Dropout-20	[-1, 128]	0
Linear-21	[-1, 128]	16,512
ReLU-22	[-1, 128]	0
Dropout-23	[-1, 128]	0
Linear-24	[-1, 2]	258
Softmax-25	[-1, 2]	0

Glossary

antigen processing (AP) A process on cellular level where random fragments of protein go through several steps to be bound to MHC and eventually be presented on the cell surface. 2, 4, 13

BA Binding Affinity: the level of binding strength between two molecules expressed in a concentration (in nm) of the IC50, often determined by competitive binding assays. The lower the concentration, the stronger the binding score. 4, 5, 13, 14

CNN CNN: Convolutional Neural Networks, Deep neural networks that utilize kernels to perform convolutions (derived from the field of Image Analysis) on the input data in order to obtain many different low level features (e.g. a circle) and eventually high level features (e.g. a face) deeper in the network. 1

Deep Learning Deep Learning: a field within Machine Learning, uses algorithms that are described as Deep Neural Networks. 10, 14

ensembles Multiple machine learning models are integrated within a framework to produce a single output using a weighted sum or another combination function, allowing the strengths of each model to be utilized. 3

ER Endoplasmatic Reticulum: unit in the cell responsible for transport and protein folding. 2

GNN Graph Neural Network. A deep neural network for Geometric Deep Learning. It can process data with inherent geometrical properties (3D objects) translated to graphs, it learns (updates) through message passing from neighbouring nodes. 14

LSTM Long Short Term Memory: a Deep Neural Network derived from a RNN (Recurrent Neural Network) that has been optimized to learn from sequence input data, by capturing long term dependencies. 3

MLP MLP: Multilayer Perceptron Networks. The first, and most simplified form of Neural Networks. It is a self-learning algorithm using non-linear functions. It is built up of several layers of nodes and edges, in each layer all nodes are connected to nodes of the next layer. Random initial numerical properties (weights) of the edges are updated through backpropagation to minimize the error between input and output. 9, 10

MS Mass Spectrometry: a technique to measure to measure the protein composition of a sample based on the weight of molecules. 4, 5, 13, 14

TAP Transporter associated with Antigen Processing: the protein complex that transports peptides. 2

References

[1] Massimo Andreatta, Bruno Alvarez, and Morten Nielsen. "GibbsCluster: unsupervised clustering and alignment of peptide sequences". In: *Nucleic Acids Research* 45.W1 (July 2017), W458–W463. ISSN: 0305-1048. DOI: 10.1093/nar/gkx248.

[2] Dominic J. Barker et al. "The IPD-IMGT/HLA Database". In: *Nucleic Acids Research* 51.D1 (Jan. 2023), pp. D1053–D1060. ISSN: 0305-1048. DOI: 10.1093/nar/gkac1011.

[3] Michal Bassani-Sternberg and David Gfeller. "Unsupervised HLA Peptidome Deconvolution Improves Ligand Prediction Accuracy and Predicts Cooperative Effects in Peptide–HLA Interactions". In: *Journal of Immunology* 197.6 (Sept. 2016), pp. 2492–2499. ISSN: 0022-1767. DOI: 10.4049/jimmunol.1600808.

[4] Kevin Michael Boehm et al. "Predicting peptide presentation by major histocompatibility complex class I: an improved machine learning approach to the immunopeptidome". In: *BMC Bioinformatics* 20.1 (Dec. 2019), pp. 1–11. ISSN: 1471-2105. DOI: 10.1186/s12859-018-2561-z.

[5] Davide Chicco and Giuseppe Jurman. "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification". In: *BioData Mining* 16.1 (Dec. 2023), pp. 1–23. ISSN: 1756-0381. DOI: 10.1186/s13040-023-00322-4.

[6] Herman N. Eisen et al. "Promiscuous binding of extracellular peptides to cell surface class I MHC protein". In: *Proceedings of the National Academy of Sciences* 109.12 (Mar. 2012), pp. 4580–4585. DOI: 10.1073/pnas.1201586109.

[7] Vanessa Jurtz et al. "NetMHCpan-4.0: Improved Peptide–MHC Class I Interaction Predictions Integrating Eluted Ligand and Peptide Binding Affinity Data". In: *Journal of Immunology* 199.9 (Nov. 2017), pp. 3360–3368. ISSN: 0022-1767. DOI: 10.4049/jimmunol.1700893.

[8] Claus Lundegaard et al. "NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8–11". In: *Nucleic Acids Research* 36.suppl$_2$ (July 2008), W509–W512. ISSN: 0305-1048. DOI: 10.1093/nar/gkn202.

[9] Dario F. Marzella et al. "Improving generalizability for MHC-I binding peptide predictions through geometric deep learning". In: *bioRxiv* (Mar. 2024), p. 2023.12.04.569776. eprint: 2023.12.04.569776. URL: https://doi.org/10.1101/2023.12.04.569776.

[10] Dario F. Marzella et al. "PANDORA: A Fast, Anchor-Restrained Modelling Protocol for Peptide: MHC Complexes". In: *Front. Immunol.* 13 (May 2022). ISSN: 1664-3224. DOI: 10.3389/fimmu.2022.878762.

[11] Dario F. Marzella et al. "The PANDORA Software for Anchor-Restrained Peptide:MHC Modeling". In: *Computational Vaccine Design*. Springer US, 2023, pp. 251–271. ISBN: 978-1-0716-3239-0. DOI: 10.1007/978-1-0716-3239-0_18.

[12] Tomas Mikolov et al. "Efficient Estimation of Word Representations in Vector Space". In: *ArXiv e-prints* (Jan. 2013). DOI: 10.48550/arXiv.1301.3781. eprint: 1301.3781.

[13] K. Natarajan et al. "MHC class I molecules, structure and function." In: *Reviews in Immunogenetics* 1.1 (Jan. 1999), pp. 32–46. ISSN: 1398-1714. URL: https://europepmc.org/article/med/11256571.

[14] Timothy J. O'Donnell, Alex Rubinsteyn, and Uri Laserson. "MHCflurry 2.0: Improved Pan-Allele Prediction of MHC Class I-Presented Peptides by Incorporating Antigen Processing". In: *Cell Systems* 11.1 (July 2020), 42–48.e7. ISSN: 2405-4712. DOI: 10.1016/j.cels.2020.06.010.

[15] Timothy J. O'Donnell et al. "MHCflurry: Open-Source Class I MHC Binding Affinity Prediction". In: *Cell Systems* 7.1 (July 2018), 129–132.e4. ISSN: 2405-4712. DOI: 10.1016/j.cels.2018.05.014.

[16] Sinu Paul et al. "HLA Class I Alleles Are Associated with Peptide-Binding Repertoires of Different Size, Affinity, and Immunogenicity". In: *Journal of Immunology* 191.12 (Dec. 2013), pp. 5831–5839. ISSN: 0022-1767. DOI: 10.4049/jimmunol.1302101.

[17] *Personalized cancer vaccine design through 3D modelling boosted geometric learning (3D-Vac)*. https://github.com/DeepRank/3D-Vac. 2024.

[18] Poomarin Phloyphisut et al. "MHCSeqNet: a deep neural network model for universal MHC binding prediction". In: *BMC Bioinformatics* 20.1 (Dec. 2019), pp. 1–10. ISSN: 1471-2105. DOI: 10.1186/s12859-019-2892-4.

[19] D. Rademaker. *GradPose*. https://github.com/X-lab-3D/GradPose. 2023.
[20] Nicolas Renaud et al. "DeepRank: a deep learning framework for data mining 3D protein-protein interfaces". In: *Nat. Commun.* 12.7068 (Dec. 2021), pp. 1–8. ISSN: 2041-1723. DOI: 10.1038/s41467-021-27396-0.
[21] Birkir Reynisson et al. "NetMHCpan-4.1 and NetMHCIIpan-4.0: improved predictions of MHC antigen presentation by concurrent motif deconvolution and integration of MS MHC eluted ligand data". In: *Nucleic Acids Res.* 48.W1 (July 2020), W449–W454. ISSN: 0305-1048. DOI: 10.1093/nar/gkaa379.
[22] Ugur Sahin and Özlem Türeci. "Personalized vaccines for cancer immunotherapy". In: *Science* 359.6382 (Mar. 2018), pp. 1355–1360. ISSN: 0036-8075. DOI: 10.1126/science.aar7112.
[23] A. Sali and T. L. Blundell. "Comparative protein modeling by satisfaction of spatial restraints". In: *Mol. Med. Today* 1.6 (Sept. 1995), pp. 270–277. ISSN: 1357-4310. DOI: 10.1016/S1357-4310(95)91170-7.
[24] Xiaoshan M. Shao et al. "High-Throughput Prediction of MHC Class I and II Neoantigens with MHCnuggets". In: *Cancer Immunology Research* 8.3 (Mar. 2020), pp. 396–408. ISSN: 2326-6066. DOI: 10.1158/2326-6066.CIR-19-0464.
[25] The HDF Group. *Hierarchical Data Format, version 5*. URL: https://github.com/HDFGroup/hdf5.
[26] Xiangyu Wu et al. "Targeting MHC-I molecules for cancer: function, mechanism, and therapeutic prospects". In: *Molecular Cancer* 22.1 (Dec. 2023), pp. 1–17. ISSN: 1476-4598. DOI: 10.1186/s12943-023-01899-4.
[27] Yilin Ye et al. "MATHLA: a robust framework for HLA-peptide binding prediction integrating bidirectional LSTM and multiple head attention mechanism". In: *BMC Bioinformatics* 22.1 (Dec. 2021), pp. 1–12. ISSN: 1471-2105. DOI: 10.1186/s12859-020-03946-z.
[28] Hao Zhang, Ole Lund, and Morten Nielsen. "The PickPocket method for predicting binding specificities for receptors based on receptor pocket similarities: application to MHC-peptide binding". In: *Bioinformatics* 25.10 (May 2009), pp. 1293–1299. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btp137.
[29] Le Zhang et al. "ConvNeXt-MHC: improving MHC–peptide affinity prediction by structure-derived degenerate coding and the ConvNeXt model". In: *Briefings in Bioinformatics* 25.3 (May 2024), bbae133. ISSN: 1477-4054. DOI: 10.1093/bib/bbae133.
[30] Zheying Zhang et al. "Neoantigen: A New Breakthrough in Tumor Immunotherapy". In: *Front. Immunol.* 12 (Apr. 2021). ISSN: 1664-3224. DOI: 10.3389/fimmu.2021.672356.