HHS Public Access

Author manuscript

Nat Mach Intell. Author manuscript; available in PMC 2023 October 12.

Published in final edited form as:

Nat Mach Intell. 2023 August ; 5(8): 861–872. doi:10.1038/s42256-023-00694-6.

Deep neural networks predict class I major histocompatibility complex epitope presentation and transfer learn neoepitope immunogenicity

Benjamin Alexander Albert[1,2], **Yunxiao Yang**[1,2], **Xiaoshan M. Shao**[1], **Dipika Singh**[3,4], **Kellie N. Smit**[3,4], **Valsamo Anagnostou**[3,4], **Rachel Karchin**[1,2,3,5,✉]

[1]Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD, USA.

[2]Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA.

[3]The Sidney Kimmel Comprehensive Cancer Center, School of Medicine, Johns Hopkins University, Baltimore, MD, USA.

[4]Bloomberg~Kimmel Institute for Cancer Immunotherapy, School of Medicine, Johns Hopkins University, Baltimore, MD, USA.

[5]Institute for Computational Medicine, Johns Hopkins University, Baltimore, MD, USA.

Abstract

Identifying neoepitopes that elicit an adaptive immune response is a major bottleneck to developing personalized cancer vaccines. Experimental validation of candidate neoepitopes is

✉**Correspondence and requests for materials** should be addressed to Rachel Karchin. karchin@jhu.edu.

Author contributions

B.A.A. and R.K. conceived the study and performed the experiments; Y.Y. contributed to 3D visualizations and model ideas; X.M.S. curated the MANAFEST data; D.S. and K.N.S. collected the MANAFEST dataset; B.A.A. and R.K. wrote the draft manuscript; B.A.A., V.A. and R.K. revised the manuscript; R.K. supervised the research.

Competing interests

Under a licence agreement between Genentech and the Johns Hopkins University, X.M.S., R.K. and the university are entitled to royalty distributions related to the MHCnuggets technology discussed in this publication. This arrangement has been reviewed and approved by the Johns Hopkins University in accordance with its conflict-of-interest policies. V.A. has received research funding to her institution from Bristol Myers Squibb, AstraZeneca, Personal Genome Diagnostics and Delfi Diagnostics in the past 5 years. V.A. is an inventor on patent applications (63/276,525, 17/779,936, 16/312,152, 16/341,862, 17/047,006 and 17/598,690) submitted by Johns Hopkins University related to cancer genomic analyses, ctDNA therapeutic response monitoring and immunogenomic features of response to immunotherapy that have been licensed to one or more entities. Under the terms of these licence agreements, the university and inventors are entitled to fees and royalty distributions. The remaining authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-023-00694-6.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-023-00694-6.

Peer review information *Nature Machine Intelligence* thanks Reid F. Thompson and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Code availability

All code used in this study and the final trained models are provided in our public GitHub repository: https://github.com/KarchinLab/bigmhc ref. 41. Scikit-Learn v.1.0.2 was used to calculate performance metrics. Pandas v.1.4.2 and Numpy v.1.21.5 were used for data processing. SAM suite v.3.5 buildmodel and align2model were used to generate multiple sequence alignments. Matplotlib v.3.5.1, Seaborn v.0.12.2, py3Dmol v.2.0.1 and v.AlphaFold2 were used to generate figures.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

extremely resource intensive and the vast majority of candidates are non-immunogenic, creating a needle-in-a-haystack problem. Here we address this challenge, presenting computational methods for predicting class I major histocompatibility complex (MHC-I) epitopes and identifying immunogenic neoepitopes with improved precision. The BigMHC method comprises an ensemble of seven pan-allelic deep neural networks trained on peptide–MHC eluted ligand data from mass spectrometry assays and transfer learned on data from assays of antigen-specific immune response. Compared with four state-of-the-art classifiers, BigMHC significantly improves the prediction of epitope presentation on a test set of 45,409 MHC ligands among 900,592 random negatives (area under the receiver operating characteristic = 0.9733; area under the precision-recall curve = 0.8779). After transfer learning on immunogenicity data, BigMHC yields significantly higher precision than seven state-of-the-art models in identifying immunogenic neoepitopes, making BigMHC effective in clinical settings.

Class I MHC plays a crucial role in vertebrate adaptive immunity. The MHC region is highly polymorphic and comprises thousands of known alleles, each encoding a molecule with varying ligand specificities. Identifying non-self antigens that are presented by a patient's MHC molecules and elicit strong immune responses may yield precise immunotherapies[1]. Tumour-specific antigens, called neoantigens, and their antigenic determinants, neoepitopes, are valuable targets for personalized cancer immunotherapies. However, identifying neoepitopes that elicit an antigen-specific immune response is a needle-in-a-haystack problem; the number of non-immunogenic candidates far surpasses the few immunogenic ones. Because experimental validation of immunogenicity is extremely resource intensive, it is critical that the top neoepitope predictions are immunogenic. To address this challenge, we present a deep neural network ensemble called BigMHC for predicting immunogenic neoepitopes with improved precision.

Intracellular proteins are degraded by proteasomes, after which the resulting peptides may be carried by transporters associated with antigen processing (TAP) molecules to the endoplasmic reticulum. Within the endoplasmic reticulum, MHC-I molecules may bind peptides to form a peptide–MHC (pMHC) complex, which may be presented at the cell surface for T-cell receptor (TCR) recognition and subsequent CD8+ (cluster of differentiation 8 positive) T-cell expansion. The set of peptides in each stage is a superset of the following; in other words, given an MHC molecule M, then $C \supset A \supset B \supset P \supset R \supset T \supset S$, where:

- C is the set of peptides derived from proteasomal cleavage.
- A is the set of peptides transported by TAP molecules to the endoplasmic reticulum.
- B is the set of peptides that binds to M to form a pMHC complex.
- P is the set of peptides present on the cell surface.
- R is the set of peptides that forms TCR–MHC complexes.
- T is the set of peptides that elicits CD8+ T-cell clonal expansion.

- S is the set of peptides that elicits a clinically observable antigen-specific immune response.

Some prior works have explicitly incorporated set C by modelling proteasome cleavage[2–4] and set A by estimating TAP transport efficiency[4]. Classifiers of set B train on in vitro binding affinity (BA) assay data[1,2,4–11]; however, BA data do not capture the endogenous processes that yield sets C and A, so BA data capture a strict superset of set B. BA data may be qualitative readings or quantitative half-maximal inhibitory concentration (IC_{50}) data. Qualitative readings were mapped to IC_{50} values[12], and the domain of IC_{50} measurements was scaled such that the weakest BA of interest, 5×10^5 nmol l^{-1}, was mapped to 0 accordingly: $f(IC_{50}) = \max(0,\ 1 - \log_{5 \times 10^5}(IC_{50}))$.

Mass spectrometry data include naturally presented MHC ligands, referred to as eluted ligands (EL); mass spectrometry data implicitly capture sets C, A and B while explicitly representing set P. EL data provide positive training examples and random pMHC data are generated for negative training examples. Some models[1,2,6,9–11] train on both BA and EL data, whereas other models[3,13,14], including BigMHC, do not train on BA. To classify set R, a recent method[15] incorporated complementarity-determining region (CDR)3β sequences from the TCR to predict the BA between TCR and pMHC. Although TCR information may be useful for predicting immunogenicity, most current datasets do not include such data. Data for classes R, T and S were also very limited, making it difficult to train classifiers directly for these sets. Prior classifiers of set T incorporate[13,14,16] the predictions from classifiers of sets B and P. To the best of our knowledge, there are no predictors of set S.

We briefly overview seven state-of-the-art methods to which we compare the proposed BigMHC (bɪg mæk) method. NetMHCpan-4.1[6] predicts sets B and P using an ensemble of 100 single-layer neural networks. NetMHCpan introduced the idea of a pan-allelic network[7,8], which consumes a peptide and a short representation of an MHC allele of interest, thereby allowing a single model to generalize across MHC alleles. NetMHCpan-4.1, like many prior models, predicts raw scores in the range [0,1] in addition to a percentage rank output in the range [0,100], which normalizes the score to a given allele. MHCflurry-2.0[2] is a pan-allelic method that predicts sets B and P using an ensemble of multilayer feed-forward networks, convolutional networks and logistic regression. MHCflurry-2.0 optionally consumes the regions flanking the N- and C-terminals to explicitly model set C. TransPHLA[11] is a pan-allelic method that predicts set P, using a transformer-based model. MHCnuggets[1] predicts set B, using allele-specific long short-term memory (LSTM) networks. HLAthena[3] has pan-allele models that predict set P with single-layer neural networks and optionally consume transcript abundance and peptide flanking sequences. MixMHCpred[9,10] predicts set P, using a mixture model and position weight matrices to extract epitope motifs. PRIME[13,14] is an extension of MixMHCpred to predict set T and was designed to infer the mechanisms of TCR recognition of pMHC complexes.

Using the procedure illustrated in Fig. 1, we developed two BigMHC models: BigMHC EL and BigMHC IM. To predict set P, BigMHC EL trains on EL mass spectrometry data and random negatives. Then, using BigMHC EL as a base model, BigMHC IM transfer learns directly on immunogenicity data to predict set T. Because $P \supset T$, transfer learning narrows

the original classification task rather than transferring to an entirely new one. Transfer learning was performed by retraining the final and penultimate fully connected layers of the base model on immunogenicity data.

Each BigMHC network model (Fig. 2a) comprises over 87 million parameters, totalling about 612 million parameters in the ensemble of seven networks. The architecture is designed to capture recurrent patterns via a wide, dense, multilayered, bidirectional LSTM (BiLSTM) and pMHC anchor site binding information via an anchor block. The BiLSTM cells are preceded by self-attention modules; these units are equivalent to transformer multi-headed attention modules[17] where the number of heads is set to one. Each wide BiLSTM cell unroll, illustrated in Fig. 2b, consumes the entire MHC representation while recurrently processing the variable-length epitope. Although this imposes a minimum epitope length of eight, few peptides of length seven or less are presented[3]. The MHC representations are novel pseudosequences generated from multiple sequence alignment; the 30 positions with highest information content are chosen to represent each allele. These positions are one-hot encoded based on the residues present at the given position, with probabilities of occurrence illustrated in Fig. 2c. The anchor block consumes the MHC pseudosequence along with the first and last four residues of the peptide to focus on the anchor-site residues. The anchor block comprises two dense[18] linear layers with tanh activations, followed by Dropout[19] units with a probability of 0.5. The outputs of the BiLSTM and the anchor block are concatenated before being consumed by a pre-attention block, which also comprises two dense linear layers with tanh activations, proceeded by Dropout with a probability of 0.5. The output is projected to the same size as the MHC one-hot encoding and passed through tanh activation to attend to the MHC encoding. This attention vector can then be superimposed onto a three-dimensional structure of an MHC allele of interest to identify important amino acid residue positions for a given pMHC, as illustrated in Extended Data Figs. 1 and 2. Moreover, because the final output of the model is a linear combination of the MHC one-hot encoding, the scalar output is interpretable, each MHC position is assigned a weight that contributes in favour of, or against, presentation, and their sum is the model output prior to sigmoid activation.

We compared the features of BigMHC with those of seven state-of-the-art methods, as shown in Table 1. We also included information on whether the models are retrainable, open-source, offer GPU acceleration, minimum and maximum peptide length, allow additional context such as flanking sequences or gene expression data, webserver availability and peptide amino acid restrictions.

Results

Epitope presentation prediction

BigMHC, NetMHCpan-4.1, TransPHLA, MixMHCpred-2.1 and MHCnuggets-2.4.0 were first evaluated on a set of 45,409 EL (set *P*) and 900,592 random decoys serving as negatives. This dataset is the same as that used to evaluate NetMHCpan-4.1[6], but with 140 deduplicated instances. Some prior methods[2,3,13,14] could not be evaluated on this dataset as they trained on the EL or do not predict set *P*.

The results, illustrated in Fig. 3a, suggest that BigMHC improves EL predictive capability, reaching 0.9733 mean area under the receiver operating characteristic (AUROC) and 0.8779 mean area under the precision-recall curve (AUPRC) when stratifying by MHC. The best prior method was NetMHCpan-4.1 ranks, with 0.9496 mean AUROC and 0.8329 mean AUPRC. The distributions of AUROC and AUPRC across human leukocyte antigen (HLA) loci are illustrated for each classifier in Fig. 3b. BigMHC demonstrates strong performance across HLA loci, whereas the performance of other methods degrades slightly on HLA-A, and particularly HLA-C. The median positive predictive value among the top n outputs (PPVn) across alleles, as previously calculated[6], for each method are BigMHC (0.8617), NetMHCpan-4.1 ranks and scores (0.8279), MixMHCpred-2.1 ranks (0.7907), MixMHCpred-2.1 scores (0.7898), TransPHLA (0.6839) and MHCnuggets-2.4.0 (0.6507).

We further stratify by both MHC and peptide length, as illustrated in Fig. 3c. After applying this more granular stratification, BigMHC yields mean AUROC and AUPRC of 0.9290 and 0.6132, respectively. By comparison, NetMHCpan-4.1 yielded mean AUROC and AUPRC of 0.8544 and 0.5266, respectively. BigMHC is most effective for peptides of length nine, which is the most common length of peptides presented by MHC-I[3,10]. Although the predictive capability of BigMHC decreases as the peptide length increases, it is still superior to that of the compared methods for all peptide lengths. Overall, BigMHC achieves higher AUROC and AUPRC than these prior methods across both types of stratifications. The two-tailed Wilcoxon signed-rank tests illustrated in Fig. 3d suggest that the BigMHC improvements are statistically significant (adjusted $P < 0.05$) after Bonferroni correction across the number of compared predictors. Tabular results are provided in Source Data Fig. 1.

Immunogenicity prediction

The vast majority of neoepitopes are not immunogenic. Furthermore, experimental validation of immunogenicity currently is not high throughput, so it is necessary to select a short list of candidate neoepitopes that can be validated per patient in a clinical setting. It is therefore critical that predictors of immunogenic neoepitopes have high precision among their most highly ranked outputs, as only the top predictions are used in practice. To measure this precision, it is common to evaluate PPVn[1,2,6]. To calculate PPVn, the pMHCs are first sorted by a predictor's output. Then, PPVn is the fraction of the top n pMHCs that are actually immunogenic.

Evaluation of immunogenicity prediction is conducted on two independent datasets: one comprising neoepitopes and the other comprising infectious disease antigens. The precision of predicting immunogenic neoepitopes is shown in Fig. 4a; we plot PPVn against all choices of n such that a perfect predictor yields a PPVn of one. This shows that the top nine predictions are all immunogenic, and as the number of predictions increases, the fraction of predictions that are actually immunogenic remains well above the PPVn of prior methods for all n. To summarize this PPVn curve, the mean PPVn is plotted with 95% confidence interval (CI) whiskers in Fig. 4c, showing that BigMHC IM achieves a mean PPVn of 0.4375 (95% CI: [0.4108, 0.4642]), significantly improving over the best prior method, HLAthena ranks, which achieves a mean PPVn of 0.2638 (95% CI:

[0.2572, 0.2705]). Additionally, these data demonstrate the utility of transfer learning to the immunogenicity domain as BigMHC IM significantly outperforms BigMHC EL, which achieves mean PPVn of 0.2704 (95% CI: [0.2632, 0.2776]). A third BigMHC curve is plotted, called BigMHC ELIM, for which we use BigMHC IM predictions on HLA-A and HLA-B peptides and BigMHC EL predictions for HLA-C peptides. Because there were very few HLA-C instances on which to transfer learn, we hypothesized that BigMHC IM may struggle with HLA-C prediction. BigMHC ELIM improved neoepitope immunogenicity AUROC and AUPRC, but did not improve on the infectious disease dataset; this is likely due to the neoepitope dataset being enriched in negative HLA-C peptides compared with the infectious disease dataset, as seen in Extended Data Fig. 5. BigMHC IM and BigMHC ELIM mean PPVn is not significantly different for neoepitope immunogenicity and slightly degrades (adjusted $P < 0.05$) for infectious disease immunogenicity as determined by two-tailed Wilcoxon signed-rank test with Bonferroni correction. Immunogenicity prediction results stratified by epitope length are presented in Extended Data Fig. 3 for neoepitopes and Extended Data Fig. 4 for infectious disease epitopes.

Precision curves and the corresponding mean PPVn for the infectious disease antigen dataset is illustrated in Fig. 4b,d. BigMHC IM achieves a mean PPVn of 0.7999 (95% CI: [0.7980, 0.8018]). The best prior method, PRIME-2.0 scores, achieves a mean PPVn of 0.7991 (95% CI: [0.7967, 0.8015]). The two-tailed Wilcoxon signed-rank test suggests that the difference between BigMHC IM and PRIME-2.0 scores is asymmetric about zero (adjusted $P < 0.05$), but because BigMHC IM just barely improves over PRIME-2.0 precision, we consider these two methods comparable for infectious disease immunogenicity prediction precision.

In addition to precision, we also report AUROC and AUPRC for each dataset along with 1,000-fold bootstrapped 95% CIs. The AUROC scores for the neoepitope and infectious disease immunogenicity datasets are reported in Fig. 4e,f and the AUPRC scores are in Fig. 4g,h. MHCnuggets-2.4.0 achieved the highest AUROC on the neoepitope dataset at 0.5852 (95% CI: [0.5833, 0.5862]) and BigMHC ELIM achieved the next best AUROC at 0.5736 (95% CI: [0.5721, 0.5750]). BigMHC ELIM significantly outperformed all prior methods on neoepitope AUPRC, reaching a mean AUPRC of 0.3234 (95% CI: [0.3216, 0.3253]), whereas the best prior method, NetMHCpan-4.1 scores, yielded a mean AUPRC of 0.2462 (95% CI: [0.2441, 0.2483]). BigMHC IM yielded AUROC of 0.5348 (95% CI: [0.5332, 0.5363]) and AUPRC of 0.3147 (95% CI: [0.3129, 0.3165]) on the neoepitope dataset, whereas BigMHC EL yielded a mean AUROC of 0.5264 (95% CI: [0.5249, 0.5280]) and AUPRC of 0.2415 (95% CI: [0.2401, 0.2428]), further demonstrating significant improvement after transfer learning.

On the infectious disease antigen dataset, PRIME-2.0 scores achieved the best AUROC and AUPRC, reaching 0.5953 (95% CI: [0.5940, 0.5966]) and 0.7905 (95% CI: [0.7893, 0.7916]), respectively. BigMHC ELIM achieved the next best AUROC at 0.5876 (95% CI: [0.5863, 0.5890]) and BigMHC IM achieved the next best AUPRC at 0.7869 (95% CI: [0.7856, 0.7882]), though both BigMHC IM and BigMHC ELIM yielded similar AUROC and AUPRC on this dataset. As with the neoepitope dataset, both BigMHC IM and BigMHC ELIM improved AUROC and AUPRC over BigMHC EL. The best AUROC and AUPRC for both immunogenicity datasets are statistically higher (adjusted $P < 0.05$) than the next best

as suggested by two-tailed Wilcoxon signed-rank tests with Bonferroni corrections. Tabular results are provided in Source Data Fig. 2.

MHC attention

The BigMHC network architecture offers a unique attention mechanism whereby prior to sigmoidal activation, the scalar output of the network is a linear combination of the input MHC encoding. Hence, we were able to visualize interpretable attention, the amino acid residue positions important for classification, in the form of a heatmap overlay on a modelled three-dimensional structure of an MHC molecule of interest. The mean attention for each pseudosequence position per allele in the EL evaluation dataset is illustrated in Extended Data Fig. 1a. The MHC molecules from each HLA locus that yielded the highest AUPRC are visualized with attention colouring in Extended Data Fig. 1b using py3Dmol[20] and AlphaFold[21] to generate MHC protein structure models. The EL training set attention values are visualized in Extended Data Fig. 2. The proposed MHC pseudosequences are comprised of the top 30 aligned positions from a cross-species alignment of 18,929 MHC-I sequences by information content; the most important are those that are in the binding groove. For certain alleles, however, some transmembrane and intracellular residues strongly contribute to EL prediction, such as position 320 for HLA-C*07:02 and 329 for many HLA-B alleles. This suggests that the NetMHCpan pseudosequences, which capture positions only nearest to the peptide, may lose information valuable for predicting pMHC presentation. Importantly, this affects all the referenced pan-allele state-of-the-art methods as they currently adopt NetMHCpan pseudosequence MHC representations.

Discussion

We first trained BigMHC to predict peptide presentation (set *P*) because an enormous amount of EL mass spectrometry data for MHC class I peptide presentation is publicly available, making it feasible to train deep learning models with over 87 million parameters. BigMHC EL achieved the highest predictive capability for set *P*, significantly outperforming the four compared methods across HLA loci and epitope lengths. We further demonstrated several technical findings: training on pMHC BA is unnecessary for predicting pMHC presentation, information content is a useful approach for deriving new MHC pseudosequence representations, and some transmembrane and intracellular MHC positions may be important for presentation prediction.

While the goal of neoepitope prediction is ultimately to predict neoepitopes that induce a clinically observable antigen-specific immune response (set *S*), there is limited immunogenicity data to train deep learning models. To address the data scarcity problem, after initially training the base models on presentation data (set *P*), we applied transfer learning using immunogenicity data (set *T*) to produce BigMHC IM. We evaluated BigMHC IM and seven other methods on two independent datasets: neoepitope immunogenicity and infectious disease antigen immunogenicity. We demonstrated strong precision on both datasets, but particularly outperformed all prior methods on the neoepitope immunogenicity prediction. We suspect that BigMHC IM outperforms other tools on neoepitope PPVn but performs similarly on the infectious disease set because of the composition of the training

data used for transfer learning. This data is a mixture of neoepitopes, cancer–testis antigens, and viral antigens. The neoepitopes represent a majority of the examples; out of 6,873 experimentally validated examples, 5,279 are neoepitopes.

However, there are several notable limitations to this study. Firstly, our evaluation of presentation prediction is based on eluted MHC ligands detected by mass spectrometry, but the negative data are random. Hence, positive data are limited to the detection efficiency of mass spectrometry, and negative data are not guaranteed negative. Two alleles, namely HLA-B*07:02 and HLA-C*03:03, yielded slightly lower AUPRC than other alleles; we suspect that differences in allele performance are primarily caused by differing class imbalances across the peptide length distributions. Another limitation is that BigMHC can only operate on MHC class I data, whereas some other methods[1,6] can predict both MHC-I and MHC-II presentation. Although we compare against state-of-the-art methods, there are many other such tools that are not compared here, and EL evaluation could not include MHCflurry-2.0, MixMHCpred-2.2, and HLAthena as their training data included most, or all, of the presented epitopes. The datasets used in this study do not have contextual information such as epitope flanking sequences and gene expression data, which may improve MHCflurry-2.0 and HLAthena results. We note that we could not compare performance in a leave-one-allele-out cross-validation experiment as NetMHCpan-4.1, MixMHCpred-2.1, and TransPHLA are not retrainable, and training BigMHC is computationally expensive. We note that a major limitation of our pseudosequences is that new alleles cannot be added without needing to retrain BigMHC from scratch; adding new alleles likely changes the resulting multiple sequence alignment, thereby affecting all other pseudosequences. We also could not answer the question as to whether BigMHC IM could discriminate between immunogenic neoepitopes and presented non-immunogenic neoepitopes; such an experiment would require pMHCs to be validated both in immunogenicity assays and mass spectrometry assays, and to our knowledge there is no such dataset available. Lastly, although our study emphasizes the importance of achieving high precision of immunogenicity in the top-ranked predictions, all evaluations were retrospective.

Future work will include prospective evaluation of the predictions of BigMHC with neoepitope immunogenicity assays. We are implementing BigMHC in ongoing computational analyses of mutation-associated neoepitope evolution under the selective pressure of immune checkpoint blockade in neoadjuvant clinical trials of patients with non-small cell lung cancer (NCT02259621), mesothelioma (NCT03918252) and gastro-oesophageal cancer (NCT03044613).

Methods

Datasets

Epitope presentation training.—Presentation training data spanned 149 alleles and included 288,032 EL and 16,739,285 negative instances. The training dataset was compiled from the NetMHCpan-4.1[6] and MHCflurry-2.0[2] single-allelic EL datasets. These instances were split into training (positive = 259,298; negative = 15,065,287), and validation (positive = 28,734; negative = 1,673,998).

Epitope presentation evaluation.—The presentation evaluation set spanned 36 HLA alleles and comprised 45,409 EL and 900,592 negative data. The test set is the same single-allelic EL dataset as that used in the NetMHCpan-4.1 study but with 140 deduplicated instances. The pMHC instances that existed in both EL training and EL testing were removed from the training dataset.

Immunogenicity training.—BigMHC IM transfer learned on PRIME-1.0[13] and PRIME-2.0[14] datasets. The original training data consisted of viral antigens, cancer–testis antigens, neoepitopes and 9-mer peptides randomly sampled from the human proteome to supplement the negative instances. BigMHC transfer learned only on the nonrandom pMHC data, of which 1,580 are positive and 5,293 are negative. The data were split into training (positive = 1,407; negative = 4,778), and validation (positive = 173; negative = 515).

Infectious disease antigen immunogenicity evaluation.—Infectious disease antigen immunogenicity prediction was evaluated using data collected from the Immune Epitope Database (IEDB)[22] on 19 December 2022. The queries to IEDB included linear peptides, T-cell assays, MHC-I restriction, human hosts and infectious diseases. Data were further processed to allow all prior methods to be evaluated: only peptides of length at least 8 and at most 11 were kept so that HLAthena could be evaluated, peptides with dummy amino acid ‘X’ were removed as many prior methods cannot handle dummy amino acids, and pMHCs with MHC alleles incompatible with MixMHCpred and PRIME were removed. After removing the intersection with all other pMHC data, a total of 1,701 immunogenic and 644 non-immunogenic infectious disease antigens were collected.

Neoepitope immunogenicity evaluation.—The neoepitope immunogenicity dataset was compiled using NEPdb[23] downloaded on 18 December 2022, Neopepsee[24], TESLA[16], and data collected from 16 cancer patients using the MANAFEST assay[25]. NEPdb is a database of neoepitopes curated from the literature with a semi-automated pipeline, whereas Neopepsee aggregated neoepitopes from two prior sources. TESLA validated immunogenicity of neoepitope predictions from a global consortium. The MANAFEST data comprised 167 immunogenic and 672 non-immunogenic neoepitopes. MANAFEST quantifies antigen-specific T-cell clonotype expansion in peptide-stimulated T-cell cultures. After removing the intersection with all other pMHC data, a total of 198 immunogenic and 739 non-immunogenic neoepitopes were collected. As with the infectious disease antigen immunogenicity dataset, the only peptides kept were of length at least 8 and at most 11 so that HLAthena[3] could be evaluated and peptides with dummy amino acid ‘X’ were removed.

MANAFEST data collection.—The MANAFEST neoepitope data were collected and processed using an established protocol[25–27]. It was generated from functional experiments of mutation-associated neoantigen-stimulated autologous T-cell cultures for 16 patients with non-small-cell lung cancer. Neoantigen-specific T cells were identified in peripheral blood using the MANAFEST assay as previously described[25–27]. For each case, tumour whole exome sequencing data were utilized to determine non-synonymous mutations and mutation-associated neoantigen candidates matched to each patient’s MHC class I haplotypes were computed as previously described[28]. ImmunoSELECT-R software from Personal

Genome Diagnostics[26,29] was used to select putative neoantigens and the neopeptides were synthesized by JPT Peptide Technologies. ImmunoSELECT-R incorporates several tools, including predicted MHC class I affinity from NetMHCpan 3.0[5], cytoxic T lymphocyte epitope prediction from NetCTLpan[30], and average gene expression in the Cancer Genome Atlas non-small-cell lung cancer[26,29]. T cells were isolated from peripheral mononuclear cells for each case by negative selection (EasySep; STEMCELL Technologies) and cultured for 10 days as previously reported[25–27]. TCR Vβ next-generation sequencing utilizing DNA from cultured CD8+ cells was performed by the Johns Hopkins FEST and TCR Immunogenomics Core Facility using the Adaptive Biotechnologies hsTCRB Kit using survey-level sequencing (Adaptive Biotechnologies). Processed data files were analysed in the publicly available MANAFEST analysis web application (http://www.stat-apps.onc.jhmi.edu/FEST) to define neoantigen-specific T-cell clonotypes. Briefly, following data preprocessing, alignment and trimming, productive frequencies of TCR clonotypes were calculated. Neoantigen-specific T-cell clonotypes met the following criteria: (1) significant expansion (Fisher's exact test with Benjamini–Hochberg correction for false discovery rate, $P < 0.05$) compared with T cells cultured without peptide; (2) significant expansion compared with every other peptide-stimulated culture (false discovery rate < 0.05); (3) an odds ratio greater than five compared with all other conditions; (4) at least 30 reads in the positive well; and (5) at least twofold higher frequency than background clonotypic expansions as detected in the negative control condition[25–27].

Dataset compositions.—The compositions of all datasets are illustrated in Extended Data Fig. 5.

BigMHC training

BigMHC was developed using Python 3.9.13, PyTorch 1.13[31] and Compute Unified Device Architecture (CUDA) 11.7 on an AMD EPYC 7443P CPU with 256 GB of RAM, and four NVIDIA RTX 3090 GPUs each with 24 GB of RAM. The training data was split 9:1 into a training set and a validation set. Training used the AdamW optimizer[32,33] to minimize binary cross entropy loss. We fine-tuned the optimizer learning rate from our initial guess of 1×10^{-5} to 5×10^{-5} by maximizing AUPRC on the EL validation dataset. The other AdamW hyperparameters were set to their default values: $\lambda = 0.01,\ \beta_1 = 0.9,\ \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Seven such BigMHC EL models were trained with varying batch sizes in $\left\{2^k\ \forall k \in \left\{9,\ 10, \ldots, 15\right\}\right\}$, with the maximum batch size of 32,768 occupying all GPU memory. For each batch size, we chose the number of training epochs that maximized AUPRC on the EL validation data up to 50 maximum epochs. For the seven models trained with batch sizes 512, 1,024, …, 32,768, the best BigMHC EL epochs were, respectively: 11, 10, 14, 14, 21, 30 and 46. On the validation data, these models yielded a mean AUROC of 0.9930 and a mean AUPRC of 0.8592. Then, the EL validation set was concatenated with the EL training set and we trained a new set of seven models for the number of epochs that previously maximized AUPRC. This new ensemble was used to evaluate BigMHC EL performance on the EL testing dataset.

After evaluating EL prediction on the testing data, all EL data were merged to train a third set of seven models using the previously optimal number of EL training epochs.

This third ensemble is used as the base model for transfer learning immunogenicity. The immunogenicity training data is similarly split 9:1 for training and validation. We optimized both the batch size and number of epochs for transfer learning for each base model by choosing the batch size and epoch number that maximizes AUPRC on the immunogenicity validation data. We search all batch sizes in $\left\{2^k \ \forall k \in \left\{3,\ 4,\ 5,\ 6,\ 7\right\}\right\}$ and all epochs up to 100. In order of least to greatest base model batch size, the best BigMHC IM (batch, epoch) numbers are: (16,23), (16,23), (8,15), (64,62), (32,27), (32,31) and (64,54). On the immunogenicity validation data, these models yielded a mean AUROC of 0.7767 and a mean AUPRC of 0.5685.

MHC pseudosequence

We introduced a new MHC pseudosequence representation. Prior pan-allele methods[2,11] adopted the NetMHCpan[6] pseudosequences, which represent the MHC molecule based on residues estimated to be closest to the peptide, or used Kidera factors[3] to encode the binding pocket residues. By contrast, BigMHC uses multiple sequence alignments to identify positions with high information content. In total, 18,929 MHC-I sequences across 12 species from the IPD-IMGT/HLA[34] database, IPD-MHC 2.0[35], and UniProt[36] (accession numbers: P01899, P01900, P14427, P14426, Q31145, P01901, P01902, P04223, P14428, P01897, Q31151) were aligned using the buildmodel and align2model from the SAM suite[37–39] version 3.5 with default parameters, yielding 414 aligned residues per sequence. The top 30 positions by information content were identified using makelogo from the SAM suite and were selected to represent the MHC sequences, which can be one-hot encoded with 414 binary variables. These new pseudosequences are provided in our public Git repository.

Network architecture study

We further investigated how the two primary modules of the network architecture affect the BigMHC performance. Specifically, we studied the wide LSTM architecture and the anchor block modules. To perform this investigation, we first ablated the anchor block and evaluated this modified architecture. Then, in addition to the anchor block ablation, we reverted the Wide LSTM to the canonical implementation and evaluated the resulting model. These two studies suggest that the wide LSTM and anchor block both offer modest gains in performance. We did not ablate the LSTM because the anchor block alone is unable to differentiate between peptides of different lengths but with the same first and last four residues. Therefore, an ablated model with the anchor block alone would be unable to correctly map the input domain to the target outputs.

Anchor block.—The anchor block processes the first four and last four residues of the peptide, which we hypothesized would help BigMHC focus on the anchor site binding residues and encourage learning long-range interactions. We ablated the anchor block and, using this modified architecture, reconducted the training and evaluation protocols. When stratifying by allele and peptide length, the anchor block improved the EL AUROC by 0.0041 and EL AUPRC by 0.0058, and particularly improved on longer peptides (12–14 amino acid residues). However, these differences were not significant at the 0.05 significance level as determined by the two-tailed Wilcoxon signed-rank test. The anchor

block improved neoepitope immunogenicity mean PPVn by 0.0061 ($P = 0.046$), AUPRC by 0.0064 $\left(P = 1.6 \times 10^{-17}\right)$, but worsened AUROC by 0.0038 $\left(P = 9.7 \times 10^{-10}\right)$, as determined by the two-tailed Wilcoxon signed-rank test.

Wide LSTM.—Where a canonical LSTM implementation[1] recurrently processes a single amino acid residue per LSTM cell unroll, we increase this window so that the BigMHC Wide LSTM processes eight amino acid residues per cell unroll, overlapping each window by seven residues as illustrated in Fig. 1b. In this ablation, we compare BigMHC with the wide LSTM to BigMHC with the canonical LSTM, and neither of these implementations include the anchor block. The wide LSTM implementation may reduce burden on the LSTM cell memory management at the expense of forcing a minimum peptide length of eight. However, most methods impose this restriction, as seen in Table 1, because most peptides presented by MHC-I are at least eight amino acids in length[10], so imposing a minimum peptide length via the wide LSTM does not substantially limit BigMHC usage. Because the wide LSTM recurs seven fewer times than the canonical LSTM, the peptide interactions that it must learn are inherently shorter. The wide LSTM is also faster, improving execution speed per network on the EL test set by nearly 30%. The canonical LSTM required more memory than the wide LSTM, probably due to more cell unrolls, so the models of batch size 32,768 could not be trained. When stratifying by allele and length, the wide LSTM improved EL AUPRC by 0.0251 $\left(P = 2.5 \times 10^{-14}\right)$, and although the canonical LSTM had higher AUROC by 0.0039, that difference was not significant ($P = 0.062$) at the 0.05 significance level from the two-tailed Wilcoxon signed-rank test. For neoepitope immunogenicity prediction, the wide LSTM improved AUROC by 0.0024 $\left(P = 5.9 \times 10^{-5}\right)$, and although the wide LSTM had higher mean PPVn by 0.0021 ($P = 0.41$) and higher AUPRC by 0.0011 ($P = 0.056$), the latter two differences are not significant at the 0.05 significance level as determined by the two-tailed Wilcoxon signed-rank test.

Compared methods—NetMHCpan-4.1[6] is a popular tool for simultaneously predicting BA and EL. This method consists of an ensemble of 100 single-layer networks, each of which consumes a peptide 9-mer binding core and a subsequence of the MHC molecule. The 9-mer core is extracted by the model, whereas the MHC representation, termed a ‘pseudosequence,’ is a predetermined 34-mer core extracted from the MHC molecule sequence. The 34-mer residues were selected based on the estimated proximity to bound peptides so that only residues within 4 Å were included.

MHCflurry-2.0[2] is an ensemble of neural networks that predicts BA and EL for MHC-I. BA prediction is the output of a neural network ensemble, where each member is a two- or three-layer feed-forward neural network. Then, an antigen processing convolutional network is trained on a subset of the BA predictions, along with the regions flanking the N-terminus and C-terminus, to capture antigen processing information that is missed by the BA predictor. EL prediction is the result of logistically regressing BA and antigen processing outputs. The MHC representation was adopted from NetMHCpan pseudosequences and expanded by three residues to differentiate some HLA alleles. Although MHCflurry-2.0 and some other tools use the EL test data in their training sets, we provide the results of

user-facing versions of each tool on all EL data, regardless of train or test data overlap, in Supplementary Table 5.

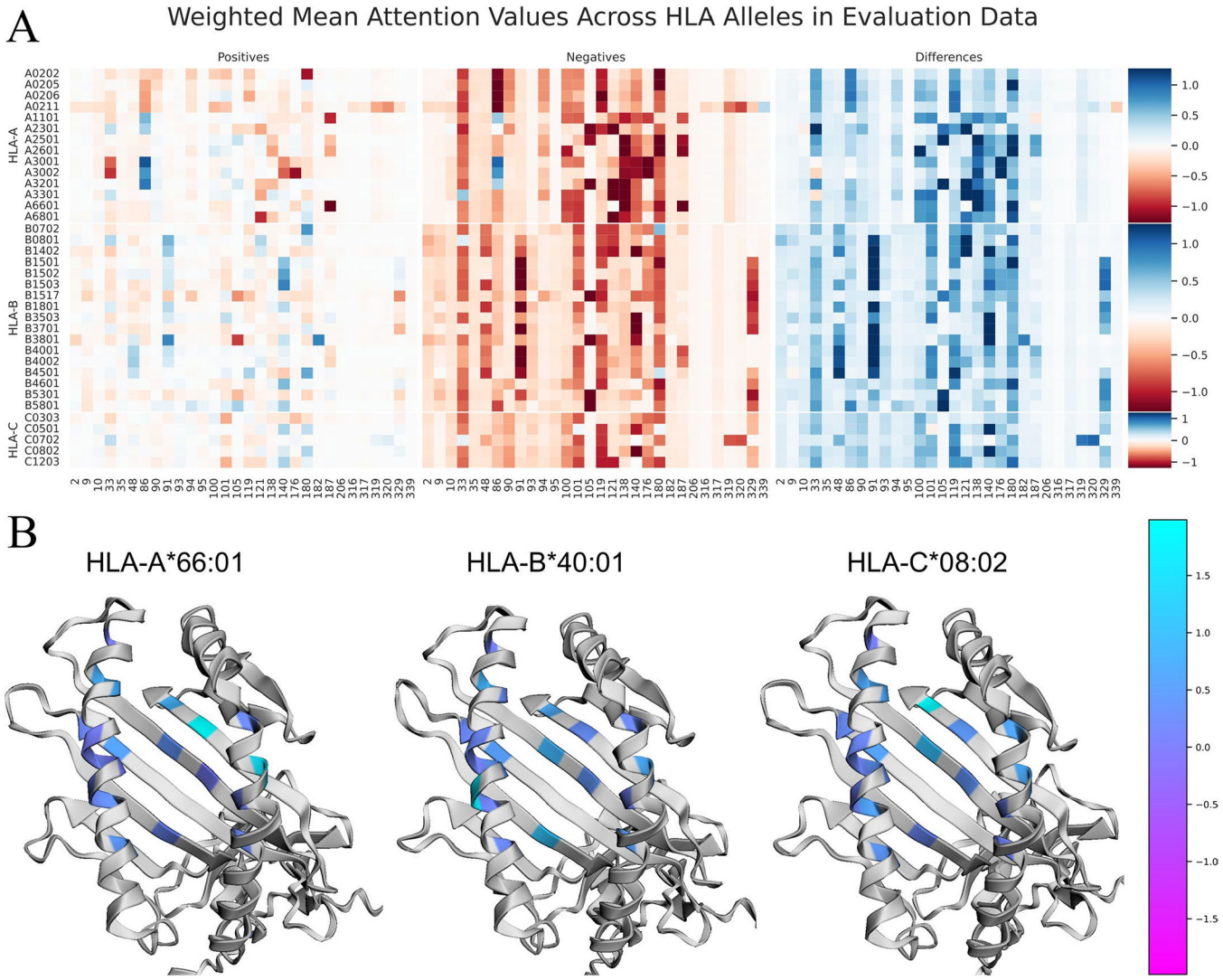
TransPHLA[11] is a transformer-based model that adopts NetMHCpan pseudosequences for MHC encoding. This model encodes peptides and MHC pseudosequences using the original transformer encoding procedure[17] before inferring the encodings via rectified linear unit (ReLU)-activated fully connected layers. TransPHLA was trained on the BA and EL data, although the BA data were binarized as binding or non-binding instances. We found that the final softmax activation of TransPHLA forces many outputs to precisely zero or one, which prevents the calculation of metrics, such as AUROC and AUPRC. Therefore, we removed the final softmax activation from TransPHLA to increase model output granularity. Because softmax is monotonic and none of the evaluation metrics rely on arbitrarily thresholding model outputs, removing the final softmax activation does not affect the evaluation metrics used in this study.

MHCnuggets[1] comprises many allele-specific LSTM networks to handle arbitrary-length peptides for MHC-I and MHC-II. Transfer learning was used across the alleles to address data scarcity. MHCnuggets trained on qualitative BA, quantitative BA, and EL data. MHCnuggets trained on up to two orders of magnitude fewer data than the other methods.

MixMHCpred[9,10,14] is built using positional weight matrices to extract epitope motifs for each allele in their training data for peptides of lengths 8 to 14. MixMHCpred-2.1 was used to evaluate EL performance because MixMHCpred-2.2 trained on the EL testing data. PRIME[13,14] builds off MixMHCpred, training directly on immunogenicity, and was designed to infer the mechanisms of TCR recognition of pMHC complexes. Upon evaluating MixMHCpred versions 2.1 and 2.2 and PRIME versions 1.0 and 2.0, both of the methods' newest versions offer substantial improvement over their predecessors.

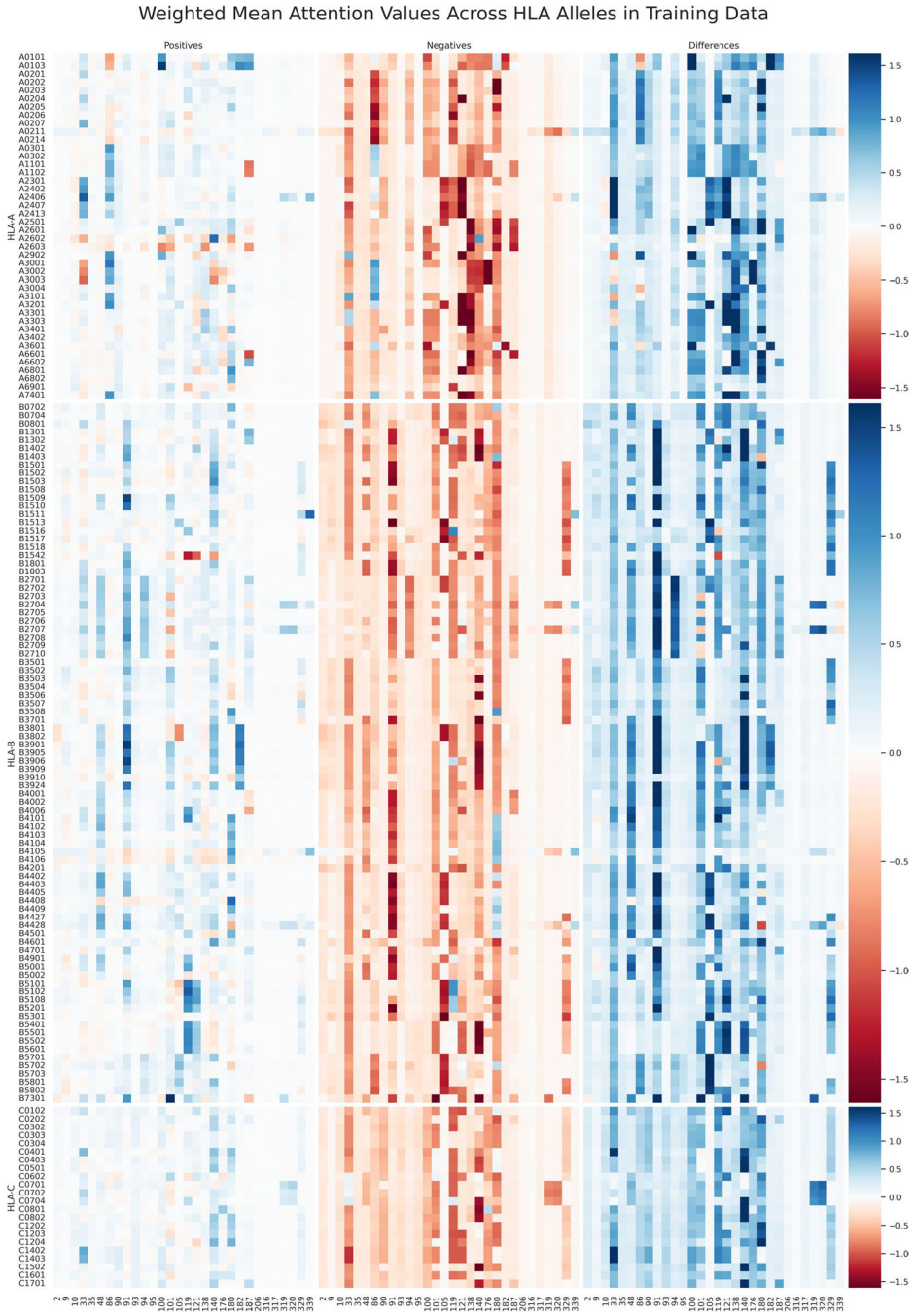
HLAthena[3] uses three single-layer neural networks trained on mass spectrometry data to predict presentation on short peptides with length in the range [8,11]. Each of the three networks trained on a separate peptide encoding: one-hot, BLOSUM62, and PMBEC[40]. In addition, the networks consumed peptide-level characteristics, and also amino acid physicochemical properties. The outputs of these networks were used to train logistic regression models that also accounted for proteasomal cleavage, gene expression and presentation bias. HLAthena also saw performance gains when considering RNA-seq as a proxy for peptide abundance.

Extended Data



Extended Data Fig. 1 |. Visualization of BigMHC average attention to MHC encodings on the EL test data.

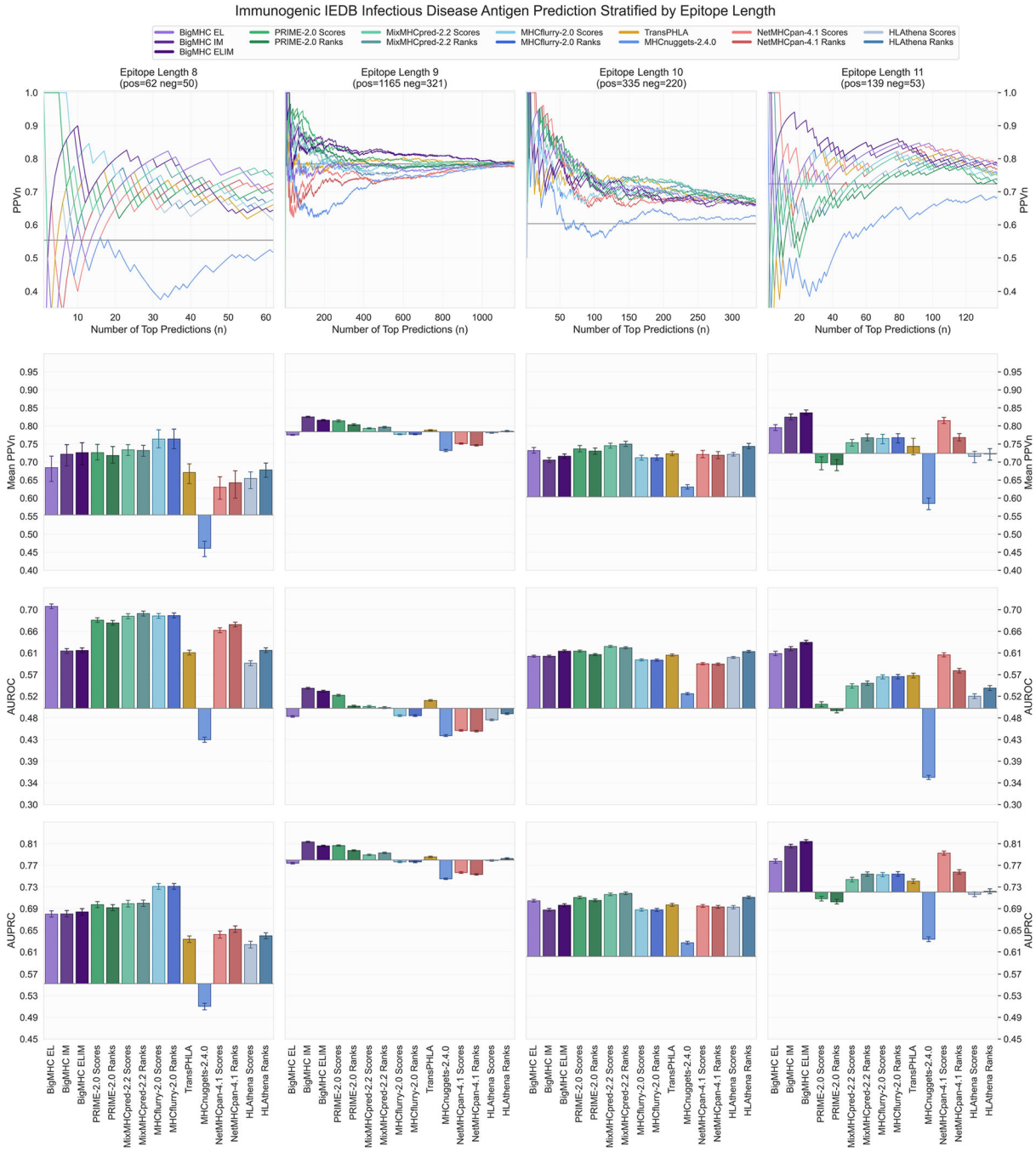
a Heatmap visualization of the average attention value for each position in the MHC pseudosequence on the EL testing dataset. The heatmap is stratified by MHC allele as rows, and separated by positive and negative testing instances. The position of each amino acid in the sequences from IPD-IMGT/HLA is provided at the bottom of each column. Darker values indicate MHC positions that are more influential on the final model output. The column of Differences depicts the Negatives values subtracted from the Positives values; thus, darker blue colours are most correctly discriminative whereas darker red attention values in this column highlight erroneous inferences. **b** Overlays of the Differences column from the training dataset on the MHC molecule using py3Dmol. MHC protein structure models are generated using AlphaFold.



Extended Data Fig. 2 |. Visualization of the average MHC attention on the EL training data. Heatmap visualization method of Extended Data Fig. 1a applied to the EL training data.

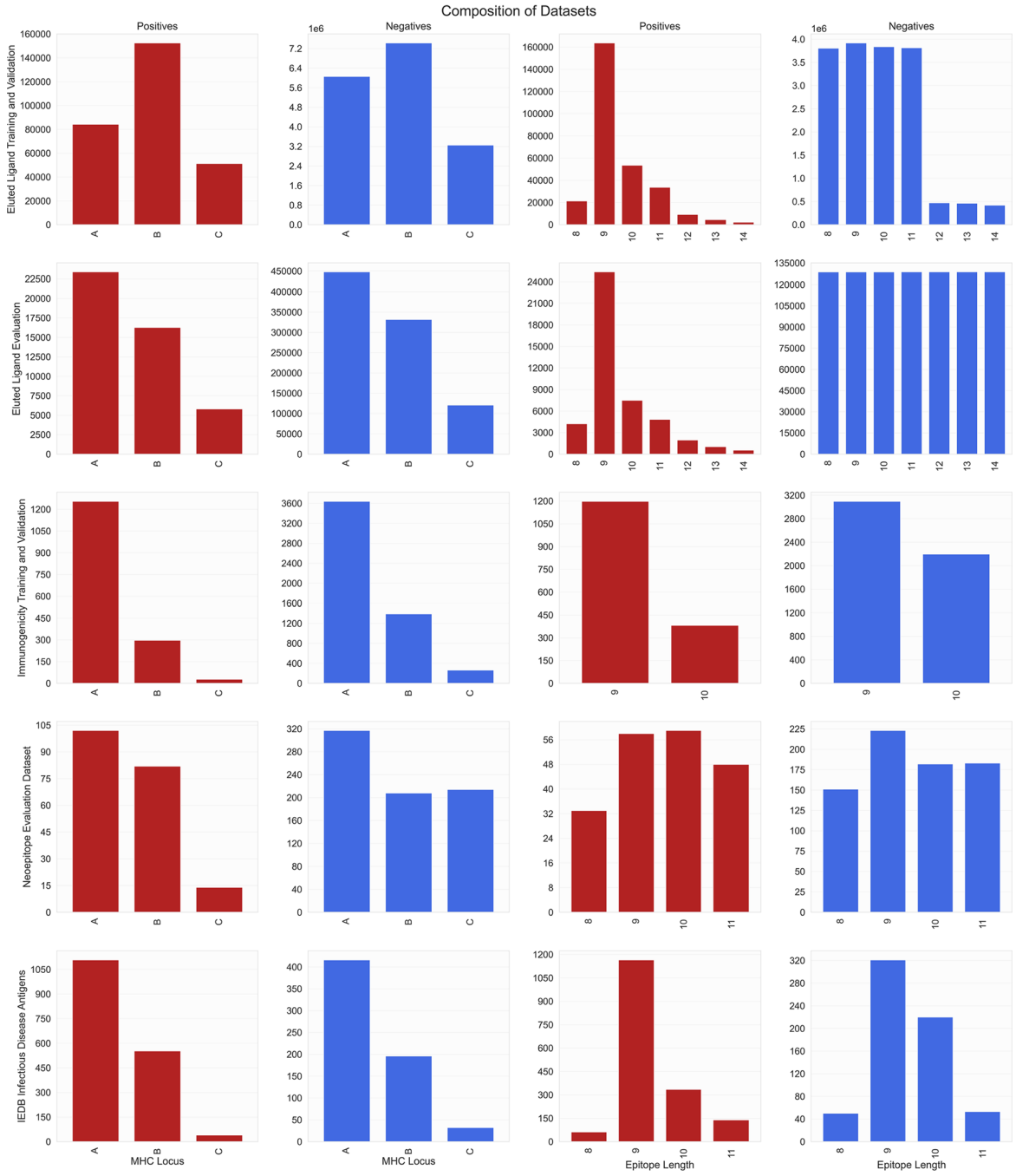
Extended Data Fig. 3 |. Neoepitope immunogenicity prediction results stratified by neoepitope length.

PPVn, mean PPVn, AUROC, and AUPRC are calculated and visualized in the same manner as Fig. 4. Bars represent means and error bars are 95% CIs. Neoepitope prediction performance from Fig. 4 is stratified by neoepitope length: 8 ($n = 184$), 9 ($n = 281$), 10 ($n = 241$), and 11 ($n = 231$).



Extended Data Fig. 4 |. IEDB infectious disease antigen immunogenicity prediction results stratified by epitope length.

PPVn, mean PPVn, AUROC, and AUPRC are calculated and visualized in the same manner as Fig. 4. Bars represent means and error bars are 95% CIs. Infectious disease antigen prediction performance from Fig. 4 is stratified by epitope length: 8 ($n = 112$), 9 ($n = 1486$), 10 ($n = 555$), and 11 ($n = 192$).



Extended Data Fig. 5 |. Composition of all training and evaluation datasets.

Positive and negative instances were stratified by HLA loci in the first two columns and by epitope length in the latter two columns. Positives in the EL datasets are detected by mass spectrometry, whereas negatives in the EL datasets are decoys. Both positives and negatives in the immunogenicity datasets are experimentally validated by immunogenicity assays.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgements

This work was supported in part by the US National Institutes of Health grant CA121113 to V.A. and R.K., the Department of Defense Congressionally Directed Medical Research Programs grant CA190755 to V.A. and the ECOG-ACRIN Thoracic Malignancies Integrated Translational Science Center grant UG1CA233259 to V.A.

Data availability

All data, including model outputs and MANAFEST data, are provided in our public Mendeley repository: https://data.mendeley.com/datasets/dvmz6pkzvb. All data except MANAFEST data were collected from publicly available sources: MHCflurry-2.0[2], NetMHCpan-4.1[6], PRIME-1.0[13], PRIME-2.0[14], TESLA[16], IEDB[22], NEPdb[23], Neopepsee[24], IPD-IMGT/HLA[34], IPD-MHC 2.0[35] and UniProt[36] (accession numbers: P01899, P01900, P14427, P14426, Q31145, P01901, P01902, P04223, P14428, P01897, Q31151). Source data are provided with this paper.

References

1. Xiaoshan SM et al. High-throughput prediction of MHC class I and II neoantigens with MHCnuggets. Cancer Immunol. Res. 8, 396–408 (2020). [PubMed: 31871119]
2. O'Donnell TJ, Rubinsteyn A & Laserson U MHCflurry 2.0: improved pan-allele prediction of MHC class I-presented peptides by incorporating antigen processing. Cell Syst. 11, 42–48 (2020). [PubMed: 32711842]
3. Sarkizova S et al. A large peptidome dataset improves HLA class I epitope prediction across most of the human population. Nat. Biotechnol. 38, 199–209 (2020). [PubMed: 31844290]
4. Stranzl T, Larsen MV, Lundegaard C & Nielsen M NetCTLpan: pan-specific MHC class I pathway epitope predictions. Immunogenetics 62, 357–368 (2010). [PubMed: 20379710]
5. Nielsen M & Andreatta M NetMHCpan-3.0; improved prediction of binding to MHC class I molecules integrating information from multiple receptor and peptide length datasets. Genome Med. 8, 33 (2016). [PubMed: 27029192]
6. Reynisson B, Alvarez B, Paul S, Peters B & Nielsen M 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. Nucleic Acids Res. W1, 48 (2020).
7. Hoof I et al. NetMHCpan, a method for MHC class I binding prediction beyond humans. Immunogenetics 61, 1–13 (2009). [PubMed: 19002680]
8. Nielsen M et al. NetMHCpan, a method for quantitative predictions of peptide binding to any HLA-A and -B Locus protein of known sequence. PLoS One 2, e796 (2007). [PubMed: 17726526]
9. Bassani-Sternberg M et al. Deciphering HLA-I motifs across HLA peptidomes improves neo-antigen predictions and identifies allostery regulating HLA specificity. PLoS Comput. Biol. 13, e1005725 (2017). [PubMed: 28832583]
10. Gfeller D et al. The length distribution and multiple specificity of naturally presented HLA-I ligands. J. Immunol. 201, 3705–3716 (2018). [PubMed: 30429286]
11. Chu Y et al. A transformer-based model to predict peptide–HLA class I binding and optimize mutated peptides for vaccine design. Nat. Mach. Intell. 4, 300–311 (2022).
12. O'Donnell TJ et al. MHCflurry: open-source class I MHC binding affinity prediction. Cell Syst. 7, 129–132.e124 (2018). [PubMed: 29960884]
13. Schmidt J et al. Prediction of neo-epitope immunogenicity reveals TCR recognition determinants and provides insight into immunoediting. Cell Rep. Med. 2, 100194 (2021). [PubMed: 33665637]
14. Gfeller D et al. Improved predictions of antigen presentation and TCR recognition with MixMHCpred2.2 and PRIME2.0 reveal potent SARS-CoV-2 CD8+ T-cell epitopes. Cell Syst. 14, 72–83.e5 (2023). [PubMed: 36603583]

15. Lu T et al. Deep learning-based prediction of the T cell receptor–antigen binding specificity. Nat. Mach. Intell. 3, 864–875 (2021). [PubMed: 36003885]
16. Wells DK et al. Key parameters of tumor epitope immunogenicity revealed through a consortium approach improve neoantigen prediction. Cell 183, 818–834 (2020). [PubMed: 33038342]
17. Vaswani A et al. Attention is all you need. In Advances in Neural Information Processing Systems (2017).
18. Huang G, Liu Z, van der Maaten L & Weinberger KQ Densely connected convolutional networks. Proc. IEEE Conference on Computer Vision and Pattern Recognition 4700–4708 (2017).
19. Srivastava N, Hinton G, Krizhevsky A, Sutskever I & Salakhutdinov R Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958 (2014).
20. Rego N & Koes D 3Dmol.js: molecular visualization with WebGL. Bioinformatics 31, 1322–1324 (2015). [PubMed: 25505090]
21. Jumper J et al. Highly accurate protein structure prediction with AlphaFold. Nature 596, 583–589 (2021). [PubMed: 34265844]
22. Vita R et al. The Immune Epitope Database (IEDB): 2018 update. Nucleic Acids Res. 47, D339–D343 (2018).
23. Xia J et al. NEPdb: a database of T-cell experimentally-validated neoantigens and pan-cancer predicted neoepitopes for cancer immunotherapy. Front. Immunol. 12, 644637 (2021). [PubMed: 33927717]
24. Kim S et al. Neopepsee: accurate genome-level prediction of neoantigens by harnessing sequence and amino acid immunogenicity information. Ann. Oncol. 29, 1030–1036 (2018). [PubMed: 29360924]
25. Danilova L et al. The Mutation-Associated Neoantigen Functional Expansion of Specific T Cells (MANAFEST) assay: a sensitive platform for monitoring antitumor immunity. Cancer Immunol. Res. 6, 888–899 (2018). [PubMed: 29895573]
26. Anagnostou V et al. Evolution of neoantigen landscape during immune checkpoint blockade in non–small cell lung cancer. Cancer Discov. 7, 264–276 (2017). [PubMed: 28031159]
27. Caushi JX et al. Transcriptional programs of neoantigen-specific TIL in anti-PD-1-treated lung cancers. Nature 596, 126–132 (2021). [PubMed: 34290408]
28. Anagnostou V et al. Multimodal genomic features predict outcome of immune checkpoint blockade in non-small-cell lung cancer. Nat. Cancer 1, 99–111 (2020). [PubMed: 32984843]
29. Jones S et al. Personalized genomic analyses for cancer mutation discovery and interpretation. Sci. Transl. Med. 7, 283ra253 (2015).
30. Stranzl T, Larsen MV, Lundegaard C & Nielsen M NetCTLpan: pan-specific MHC class I pathway epitope predictions. Immunogenetics 62, 357–368 (2010). [PubMed: 20379710]
31. Paszke A et al. PyTorch: an imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems (2019).
32. Kingma DP & Ba JL Adam: a method for stochastic optimization. In Third International Conference for Learning Representations (2015).
33. Loshchilov I & Hutter F Decoupled weight decay regularization. In Seventh International Conference for Learning Representations (2017).
34. Robinson J et al. IPD-IMGT/HLA database. Nucleic Acids Res. 48, D948–D955 (2019).
35. Maccari G et al. IPD-MHC 2.0: an improved inter-species database for the study of the major histocompatibility complex. Nucleic Acids Res. 45, D860–D864 (2016). [PubMed: 27899604]
36. Consortium TU UniProt: the universal protein knowledgebase in 2023. Nucleic Acids Res. 51, D523–D531 (2022).
37. Hughey R & Krogh A Hidden Markov models for sequence analysis: extension and analysis of the basic method. Bioinformatics 12, 95–107 (1996).
38. Karplus K et al. What is the value added by human intervention in protein structure prediction? Proteins Struct. Funct. Bioinf. 45, 86–91 (2001).
39. Krogh A, Brown M, Mian IS, Sjölander K & Haussler D Hidden Markov models in computational biology: applications to protein modeling. J. Mol. Biol. 235, 1501–1531 (1994). [PubMed: 8107089]

40. Kim Y, Sidney J, Pinilla C, Sette A & Peters B Derivation of an amino acid similarity matrix for peptide:MHC binding and its application as a Bayesian prior. BMC Bioinf. 10, 394 (2009).
41. KarchinLab/bigmhc: v1.0. Zenodo 10.5281/zenodo.8023523 (2023).

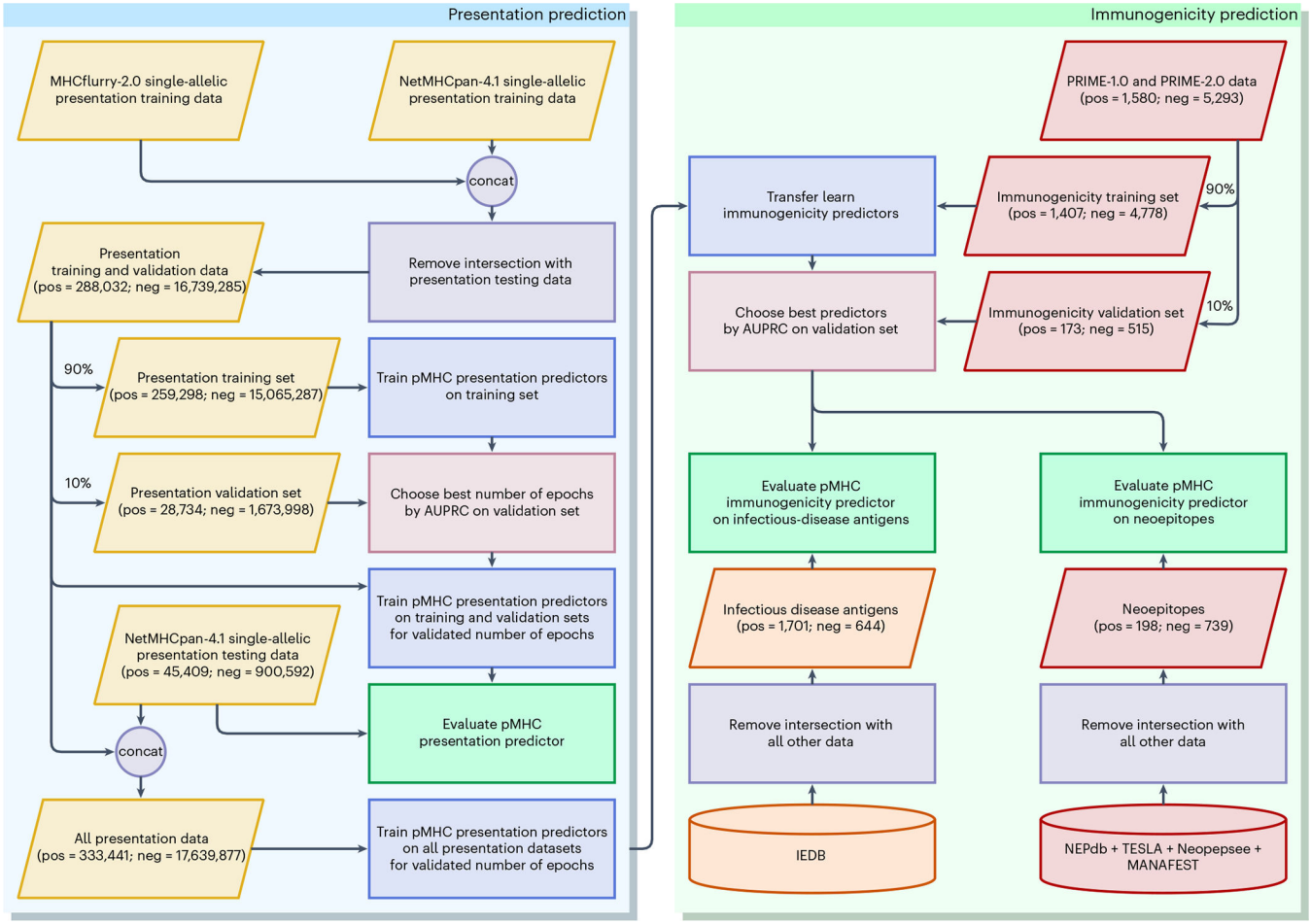


Fig. 1 |. Experimental procedure.

The procedure includes presentation training, immunogenicity transfer learning and independent evaluation on multiple datasets. The circles labelled ‘Con’ indicate dataset concatenation. Input and database symbols are color-coded by data type: presentation (yellow), immunogenicity training and neoepitope evaluation data (red), and infectious disease (orange). Rectangles are the processes: removing data overlap (purple), choosing best models (pink), training (blue), and evaluation (green).

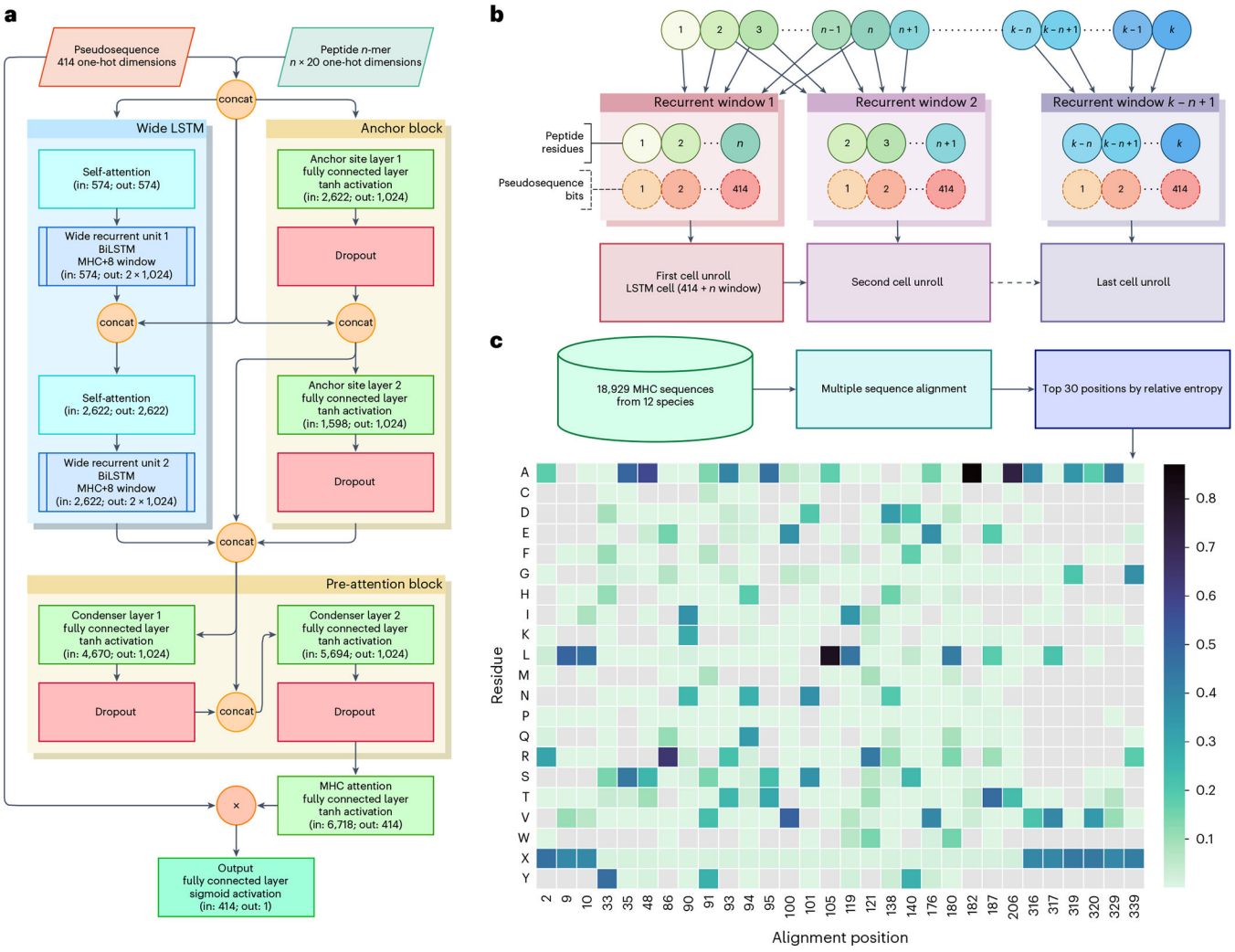


Fig. 2 |. BigMHC network architecture and pseudosequence composition.

a, The BigMHC deep neural network architecture, where the BigMHC ensemble comprises seven such networks. Pseudosequences and peptides are one-hot encoded prior to feeding them into the model. The circles labelled ‘Con’ indicate concatenation and the circle labelled ‘×’ denotes element-wise multiplication. The anchor block consists of two densely connected layers that each receive the first and last four peptide residues along with the MHC encoding. The self-attention modules are single-headed attention units, which is analogous to setting the number of heads of a standard multi-headed transformer attention module to one. Prior to the final sigmoid activation, the output of the model is a weighted sum of the MHC pseudosequence one-hot encoding; the weights are referred to as attention. Because all connections except internal BiLSTM cell connections are dense, data are not bottlenecked until the MHC attention node maps the pre-attention block output to a tensor of the same shape as the one-hot-encoded MHC pseudosequences. **b**, A wide LSTM. Each cell unroll processes the entire MHC pseudosequence but only a fixed-length window of the peptide. Where a canonical LSTM uses a window length of one, BigMHC uses a window length of eight to capitalize on the minimum pMHC peptide length. **c**, The pseudosequence amino acid residue probability (represented by the color scale) per alignment position. Note

that not all amino acid residues are present for each position, as indicated by grey cells, so the one-hot encoding uses a ragged array, encoding only the residues present at a given position.

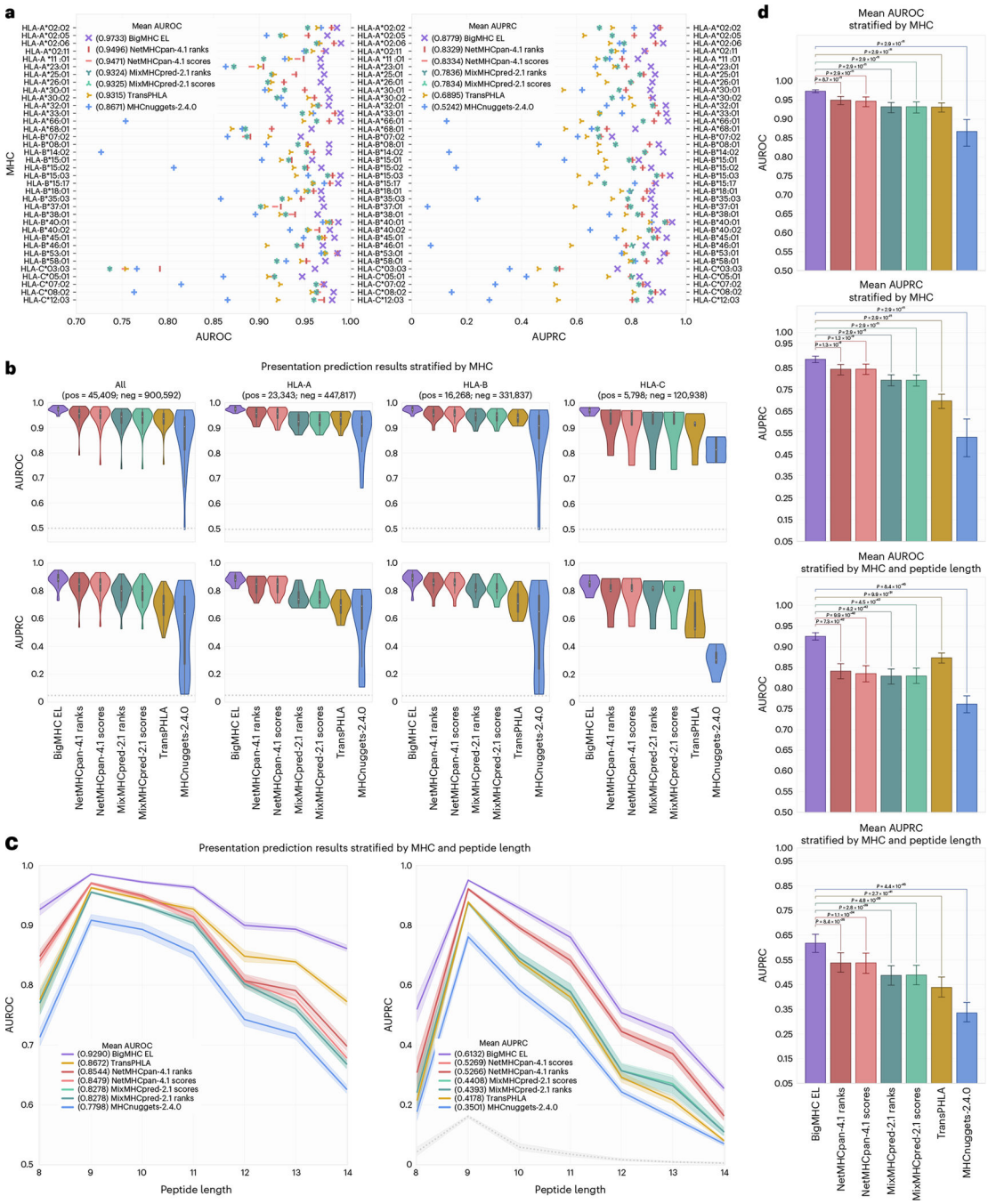


Fig. 3 |. EL prediction results.

a, AUROC and AUPRC for each allele in the EL testing dataset. **b**, AUROC and AUPRC violin plots with embedded box-and-whisker plots stratified by allele and grouped by MHC locus. **c**, Mean AUROC and AUPRC per peptide allele length with 95% CI by MHC stratification. Baseline (random) classifier performance is 0.5 for AUROC and illustrated in grey for AUPRC. **d**, Mean AUROC and AUPRC and 95% CI stratified by MHC ($n = 36$) and both MHC and epitope length ($n = 252$) with two-tailed Wilcoxon signed-rank test adjusted *P*-values across methods.

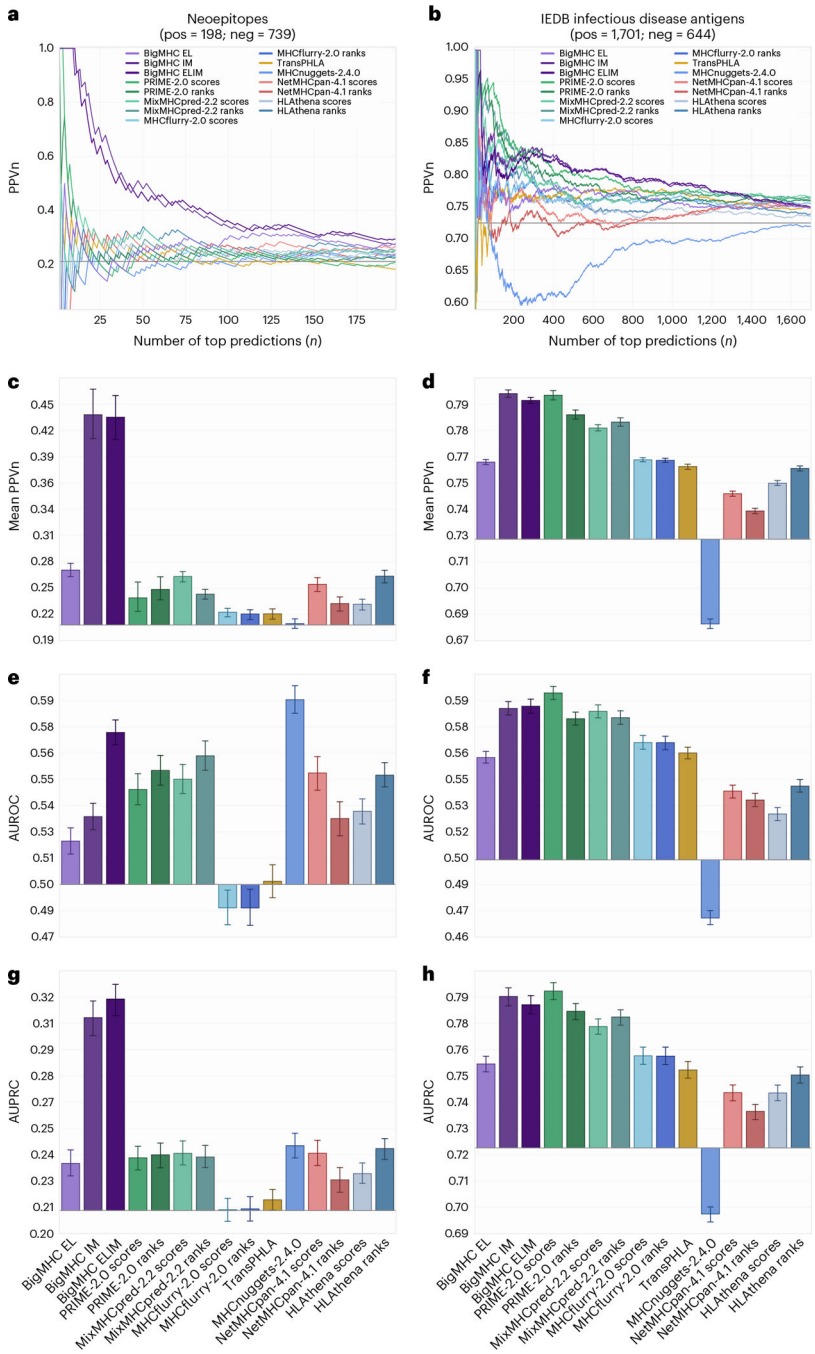


Fig. 4 |. Performance of immunogenicity predictions for all methods.

a,**b**, PPVn is calculated for each method as the fraction of neoepitopes (**a**) or infectious disease antigens (**b**) that are immunogenic within the top n predictions. **c**,**d**, The mean PPVn and 95% CI whiskers are reported for neoepitopes (**c**; $n = 937$) and infectious disease antigens (**d**; $n = 2{,}345$), summarizing the PPVn curves for all valid choices of n. The baseline PPVn, representing a random classifier, is illustrated as a horizontal line at 0.2113 for neoepitopes and 0.7254 for infectious disease antigens. **e**–**h**, Mean AUROC (**e**,**f**) and

mean AUPRC (**g**,**h**) of all methods with 95% bootstrap CIs from $n = 1{,}000$ iterations for neoepitopes (**e**,**g**) and infectious disease antigens (**f**,**h**).

Table 1 |

Method and feature comparison of BigMHC and prior works

		BigMHC	PRIME-2.0	MixMHCpred-2.2	TransPHLA	PRIME-1.0	MixMHCpred-2.1	MHCflurry-2.0	NetMHCpan-4.1	MHCnuggets-2.4.0	HLAthena
Publication year		2023	2023	2023	2022	2021	2020	2020	2020	2020	2020
Training	BA				X		X	X	X	X	
	EL	X		X	X		X	X	X	X	X
	IM	X	X			X					
Prediction	BA							X	X	X	
	EL	X		X	X		X	X	X	X	X
	IM	X	X			X					
Optional extra context								X			X
Retrainable		X						X		X	X
Transfer Learning		X									
Open source		X	X	X	X	X	X	X		X	X
Pan-allele		X			X			X	X		X
Optional single-GPU		X			X			X		X	
Optional multi-GPU		X									
Has webserver			X	X	X	X	X		X		X
Min peptide Length		8	8	8	8	8	8	5	8	None	8
Max peptide Length		None	14	14	15	14	14	15	None	None	11
Allows wild-type amino acids		X			X				X	X	

Cells with ‘X’ indicate that the method has the given feature. Training rows indicate the type of data on which models are trained, whereas prediction rows indicate what type of peptides the model explicitly predicts. Models that are provided with executables or source code for retraining on new data are considered retrainable. Pan-allele methods are those that encode the MHC sequence to generalize predictions across alleles rather than employing multiple allele-specific models. Optional extra context refers to any optional input, such as N-terminal and C-terminal flanking sequences or gene expression data. Models that can consume wild-type amino acids, are indicated in the final row. IM, immunogenicity.