NAP-CNB: Bioinformatic pipeline to predict MHC-I-restricted T cell epitopes in mice

Carlos Wert-Carvajal[1,2,3†], Rubén Sánchez-García[1†], José R Macías[1], Rebeca Sanz-Pamplona[4,5], Almudena Méndez Pérez[1], Ramon Alemany[4], Esteban Veiga[1], Carlos Óscar S. Sorzano[1] and Arrate Muñoz-Barrutia[2*]

[1]Centro Nacional de Biotecnología, Consejo Superior de Investigaciones Científicas, Calle Darwin 3, 28049 Madrid, Spain. [2]Departamento de Bioingenieria e Ingenieria Aeroespacial, Universidad Carlos III de Madrid, Av. de la Universidad 30, 28911 Leganés, Spain. [3]Bioengineering Department, Imperial College London, Exhibition Road, SW7 2AZ London, United Kingdom. [4]Catalan Institute of Oncology - IDIBELL, Av. de la Granvia de l'Hospitalet 199, 08908 L'Hospitalet de Llobregat, Spain. [5]Centro De Investigación Biomédica en Red de Epidemiología y Salud Pública (CIBERESP), Madrid, Spain.
[*]Correspondence: mamunozb@ing.uc3m.es.
[†]Equal contributor

Abstract

Motivation:
Lack of a dedicated integrated pipeline for neoantigen discovery in mice hinders cancer immunotherapy research. Novel sequential approaches through recurrent neural networks can improve the accuracy of T-cell epitope immunogenicity predictions in mice and a simplified variant selection process can reduce operational requirements.

Results
We have developed a web server tool (NAP-CNB) for a full and automatic pipeline, based on recurrent neural networks, able to predict putative neoantigens from tumoral RNA sequencing reads. The developed software is able to estimate H-2 peptide ligands, with an AUC of 0.95, directly from tumor samples. As a proof-of-concept, we used the B16 melanoma model to test the predictive capabilities of the system and we report its putative neoantigens.

Availability and implementation
NAP-CNB web server is freely available at http://biocomp.cnb.csic.es/NeoantigensApp/ with scripts and datasets accessible through the download section.

Introduction

Cancer cells can accumulate a large number of mutations that change protein sequences which can lead to MHC-restricted T-cell epitopes [1]. Identifying the tumor-specific epitopes that elicit T cell responses represents a paramount challenge for cancer immunotherapy, and in particular to design of vaccines and to detect cellular adaptive immune responses [1, 2].

A plethora of neoantigen discovery pipelines has been described to enable the prediction of epitopes from genetic information. However, current pipelines are human-centered and, thus, are primarily designed for clinical usage [3, 4]. Among the preeminent lines of research, genomic analysis adjustments [3, 5–8] and neoepitopes ranking practices [5, 6, 8, 9] have been prioritized over immunogenicity prediction algorithms. In spite of this, the latter remains a critical component of the overall workflow for which there are limited available options [10].

The absence of dedicated tools for the alternative *in vivo* mouse models hinders pre-clinical cancer immunotherapy research. Hence, laboratories have to produce or adapt to *ad-hoc* human pipelines. Solely Epi-Seq [11] and MuPeXI [9, 12] offer modified versions for the murine model. Both platforms follow the canonical prediction process, based on sequencing data to estimate the gene expression and the affinity with the T-cell receptor (TCR) affinity of the mutated peptide [10], which is a prerequisite to eliciting an immune response [1]. However, Epi-Seq is not tailored for neoantigen detection, as it was conceived for the discovery of common tumor antigens, and MuPeXI lacks genome preprocessing and variant calling in its analysis. Mainly, in both cases, the algorithms underpinning immunogenicity prediction rely on dense neural networks. In the case of Epi-Seq, it uses NetMHCPan [13], which is trained with 9meric samples from the major histocompatibility complex (MHC) of mice or H-2, whilst MuPeXI employs NetH2pan [14], that is trained with pooled sequences from a diverse set of alleles.

Supervised machine learning methods have facilitated the identification of neoepitopes. In particular, artificial neuronal networks have proven to be highly efficient [15]. However, among them, recurrent neural networks (RNN) are better suited for sequential problems, as it is attested by its extensive usage in natural language processing systems [16]. As a case, long short-term memory (LSTM) units are, at present, used for protein prediction of function and interactions [17, 18].

Prediction models have relied on gene expression information from tumoral samples to determine putative peptides for intervention [1]. However, current approaches depend additionally on genetic information from DNA sequencing to determine mutations [5, 8]. This hinders temporal performance and increases intervention costs, but whole-exome sequencing (WES) is justified for its improved selectivity [19]. Hence, a system may rely exclusively on RNA sequencing (RNA-Seq) to identify mutations and gene expression levels simultaneously [19]. If compensatory methods in immunogenicity prediction are present, a tool designed for pre-clinical use may only rely on mutational information from RNA-Seq for a cost-effective solution.

We developed an integrated pipeline optimized for a murine model that finds putative neoepitope via next-generation sequencing (NGS) tumoral variant calling and ranks them using LSTMs. This novel platform in only based on RNA-Seq and it is automatized for a given haplotype. As a proof-of-concept, we trained our system with the H-2K^b haplotype (MHC class I) for the commonly used B16 melanoma model in C57BL/6 mice.

Methods

The proposed pipeline employs genome preprocessing tools, variant calling software and customized neural network architecture to obtain putative neoantigens from RNA-Seq experiments. As an integrative tool, the workflow has been adapted into a web server for RNA-Seq file submissions (biocomp.cnb.csic.es/NeoantigensApp/). A tumor RNA-Seq file should be inputted as ".fastq.gz" together with the MHC class I type and an email address to receive the final results.

Variant calling: from RNA-Seq to mutant peptides

The somatic mutations suitable for neoantigen prediction are obtained from the gene expression of tumoral tissue (RNA-Seq). NGS technologies that produce a FASTQ file are required for this protocol.

First, a quality assessment report is produced using FastQC (v0.11.8) [20] for user evaluation. In terms of preprocessing, the RNA-Seq file is realigned with a reference genome for further processing with STAR (v2.6.0a) [21]. The resulting BAM file is processed with Picard (v2.19.2) [22] for further refinements such as annotation and duplicate marking. Subsequently, Genome Analysis Toolkit (GATK, v4.1.2.0) [23] is used for exon segmentation, through the "SplitNCigarsReads" protocol, and base recalibration in accordance with Best Practices guidelines [24]. As indicated in **Figure 1**, this part serves as a preprocessing of the RNA-Seq reads *per se* before variant calling.

The MuTect2 variant caller [25] from the GATK package is used in its tumor-only mode, as shown in **Figure 1**, which is computationally less expensive but provides a higher number of false positives [26]. Even if designed primarily for DNA-Seq reads, MuTect2 has shown to be efficient in calling mutations from RNA-Seq [27]. By default, tumoral RNA-Seq is matched with databases of single nucleotide polymorphisms (dbSNP) although it can be used with a panel-of-normals (PoN) by construction. Following depth coverage filtering, the variants are submitted to Variant Effector Predictor (VEP) from Ensembl (v100.0) [28] for annotation and extraction of mutant peptide sequences identified as missense variants. Finally, a script matches the resulting UniParc reference, from VEP to extracted UniProt proteins for protein-level prediction [29].

Additionally, Cufflinks (v2.2.1) [30] is used for mRNA abundance estimation as measured by fragments per kilobase million (FPKM). As there is no range for optimal neoantigen expression, this metric is provided to the user for its examination, as shown in **Figure 1**.

Dataset generation and preprocessing

Sequences of immunogenic peptides for algorithm development were obtained from the IEDB database [31] for the H-2K^b MHC-I haplotype of the mouse strain C57BL/6. Given the different methodologies of binding assessment considered in IEDB, elements were binarized by their MHC class I classification as positive or negative, per IEDB standards. The dataset, by entries accession number, is available at NAP-CNB.

Peptides deemed as antigenic were processed to extract their binding sites. Positive epitopes from IEDB were aligned with its protein source, through the Smith-Waterman algorithm [32], to obtain the remaining sequence as negative samples (Suppl. Fig. 1). Additionally, epitope regions were extended, through the original sequence, to have a regular size (Suppl. Fig. 1). In contrast with previous methods, a given prevalence (i.e., the fraction of the minority class) was not imposed on the dataset. In total, 4,828 peptide entries were processed into 251,049 sequences with 6,714 positive entries and 244,225 negatives. A 10% split was used for test set generation. Concerning blind test data, IEDB datasets 1034799 and 1035276 were processed through the previous procedure and by the method described by [14].

Further postprocessing was implemented with an optional majority vote algorithm that considered mutations to the most similar amino acid, given by the BLOSUM62 matrix [33], for each position. In other terms, a sequence modified its classification if there was a given consensus among its most akin peptides.

Neural network training

The neural networks were implemented through Keras (v2.2.4) [34] and TensorFlow (v1.11.0) [35]. A scalable routine was used for architecture optimization through simplified datasets (Suppl. Fig. 1) until one competent was obtained. Moreover, training was done with "on-batch" class balancing and data augmentation. The latter increased the number of positives sequences through random substitution of a given number of amino acids with similar ones from the BLOSUM62 matrix [33], with a given tolerance (Suppl. Fig. 3). Training was performed through 5-fold cross-validation, for hyperparameters tuning and optimization of balancing and augmentation, generating a total of 80 models for the actual dataset.

The initial toy model was used for embedding selection, and model tuning of neural architectures (Suppl. Table 1A-B), which was maintained in the type and depth of layers in later configurations, whilst an intermediate dataset (Suppl. Fig. 1C) was introduced for data balancing and augmentation. The final model was produced with the complete dataset and cross-validation of the number of internal LSTM units at each layer, the number of on batch sequence augmentations and its tolerance and the on batch class balancing.

In the final architecture, 12mer peptide sequences are introduced with a one-hot encoding representation to three consecutive bidirectional LSTM layers, followed by three layers of dense neurons with two intermediate dropouts units. The output layer consists of a dense neuron, with a soft-max activation, which yields the affinity estimation probability. The overall network is represented in **Figure 2** .

Sequencing raw data

An *in vitro* B16 melanoma cell line with an H-2K^b haplotype was processed for RNA extraction and sequenced through an NGS Illumina HiSeq2000. From the FastQC analysis, all evaluated parameters were satisfactory except from the presentation of four over-represented sequences corresponding to Illumina single end PCR primer and technical noise as TrueSeq adaptors. Trimming of these sequences was done before RNA-Seq processing. The resulting ".fastq.gz" file was introduced for analysis in a local server.

Results

Cross-validation metrics

Initial architectures, based on LSTM and dense layers, showed performance improvements, in terms of the area under the curve for the receiver operator characteristic (AUC ROC), for higher depth models (Suppl. Table 1A). In spite of this, these changes did not have an impact as significant as "on-batch" balancing and data augmentation. In particular, modifications of a "virtual" prevalence were able to raise AUC ROC and F-1 values to a 20% in test sets (Suppl. Table 1C) and decreased the degree of overfitting. All parameters were adjusted through grid search on the final model under a limited number of epochs (see Additional file 2 - Grid search parametrization). As observed in **Table 1,**the final AUC of the network reached 95%, albeit with an acceptable F1 score, due to the assumed low prevalence. The complete cross-validation results of each model are available at NAP-CNB.

For further evaluation, 10% of the original dataset was used as a test set of the selected parametrized system. In **Figure 3** , both the ROC and the precision-recall curve are

shown. The latter reflects how the system fares against a high class imbalance. In terms of metrics, the ROC AUC for the test sample was of 86.5% with a 97.2% accuracy. Notwithstanding, the proposed ensemble method for postprocessing is able to increase precision by 7.6%.

Throughout cross-validated models, window sizes of 8, 10 and 12 amino acids were tested for predictive performance. Sequences of 12 amino acids produced more accurate models, as observed in **Figure 4** , which may indicate that antigenic determinants are not sufficient for peptide classification and distal amino acids carry additional predictive information. The distribution of sequences classified as positive, and a sensitivity analysis from random classifications showed similar results (Suppl. Fig. 4). In the case of the typing H-2K^{d}, the best performance corresponded to 8-mers, which is the window used for mutant peptides (Suppl. Material H2-Kd). Cross-prediction between both typings was suboptimal (Suppl. Material. H2-Kd), which may indicate that neural networks should be configured for individual typing.

Benchmarking

Compared against NetH2pan [14], which is the benchmark used for MHC class I affinity prediction in mice, the reported cross-validated AUC ROC and PPV are higher for the H-2K^{b} typing. Moreover, to confirm a better performance on a dataset, blind testing was implemented from two unused H-2K^{b} datasets from IEDB (1034799 and 1035276). Negatives were generated through both the aforementioned protocol and by generating random sequences with an assumed prevalence as described in [14]. Given that NetH2Pan considers different epitope lengths and substitutions, binarization was done by considering whether binds were predicted overall for a 12mer sequence. In all binary metrics, the LSTM network achieved improved results (Suppl. Fig 5 and 6). The reported accuracies were between 96% and 98% with up to 3-fold increases in precision.

Notably, in all cases, positives were better detected than in NetH2pan, irrespective of the method used for the production of negative sequences. On the whole, our approach detected 259 and NetH2pan 86 of a total of 438 antigens.

Use case

As a result of MuTect2 calling, 4,566 variants were identified. From those, 1,085 missense transcripts were obtained from VEP corresponding to 345 genes. These were matched against the results from Cufflinks and submitted for prediction. In the end, our proposed software generated a ranking of putative neoantigens.

The thirty-five top-scoring putative neoepitopes are shown in **Table 2**. The predictions were matched with the original B16 results from Castle *et al.* [36] (Suppl. Table 2). Additionally, we compared the rank given by the softmax score of our proposed algorithm with the relative classification of the 12mer sequence in NetH2pan, obtained by averaging the scores across all of its possible epitope lengths and mutations [14]. **Table 2**, thus, shows the establishes an order of preference for both methods. Due to sample size limitations, the haplotype H-2D^{b} of the C57BL/6 model is not analyzed but should also be included in a naïve study.

In terms of overall performance, the entire pipeline has an execution time around eleven hours in a local server using two CPU cores. This duration corresponds to steps between preprocessing of the RNA-Seq and quality analysis to immunogenicity prediction. The

levels of abundance may be suboptimal in some cases but, its reporting may guide the user in the selection of a candidate.

Discussion

The proposed pipeline provides an integrated software solution for mouse neoantigen MHC class I discovery from RNA-Seq data. The workflow is based on a streamlined process that is adapted to the resource-efficient and accessibility requirements of pre-clinical research. Notably, we report an immunogenicity estimation model which successfully improves previously reported performance. The B16 case study also shows an acceptable number of putative neoantigens that is coherent with literature estimates [36]. A functional validation measuring T-cell immune responses by ELISPOT or intracellular IFN-gamma staining in mice responding B16 tumors would be required to validate the prediction results.

In terms of the actual prediction algorithm, the RNN-based approach presents an AUC of 0.95 in cross-validation. In comparison with the current NetH2pan benchmark model [14], it represents an enhancement in terms of accuracy and precision for the H-2K^b haplotype. Thus, this reinforces the suitability of sequential models as an efficient solution to antigen binding prediction against more conventional neural network approaches. Future lines of research may include more recent sequential model innovations. Novel types of sequential architectures in transformers and RNNs, such as BERT [37] and GORU [38], could serve as enhancers of overall performance. Also, subsequent work in epitope size should aim to reconcile flexibility, which is compatible with an RNN-based framework, with the generation of empirical negative samples. The webserver restricts the haplotype utilized for prediction, which can be expanded for other typings, like H-2K^d (Suppl. Material. H2-Kd). Even if cross-prediction between haplotypes K^b and K^d suggests type-specific modelling is an optimal solution, a pan-specific system is part of forthcoming directions.

With regard to data processing, the use of negative empirical sequences and data augmentation should also be considered to improve affinity estimation. Strategies could include generative models such as Gaussian mixtures or adversarial networks (GAN) [39]. Nonetheless, some of the problems posed by the dataset include its reliance on a binarized predictor, which hampers the biological meaning of the results, and the prevalence dependency of precision and recall, for which further work should be done to identify an optimal strategy. Finally, our method is characterized by the employment of window sizes that are above the normative length of an epitope to optimize performance, which may imply that reported antigenic determinants are not sufficient information for prediction. From a biological perspective, this is equivalent to regard the overall conformation of a protein as relevant in the binding process.

The variant calling process poses further challenges. Our approach has prioritized a procedure that functions solely on RNA-Seq data with a conservative selection of mutations, in particular, missense SNV. This neglects a high percentage of variants that produce neoantigens [40] and increases the mutational uncertainty by not including genomic data from DNA-Seq [19]. Advances should proceed in this direction, albeit prioritizing an exclusive RNA-Seq utilization to retain the cost-effectiveness of the tool.

Competing interests

The authors declare that they have no competing interests.

Funding
This work was partially funded by the Spanish Ministry of Economy, Industry and Competitiveness (TEC2016-28052-R, RTC2017-6600-1, SAF2017-84091-R), the Spanish Ministry of Science and Innovation (FPU18/03199", PID2019-109820RB-I00), the “la Caixa” Foundation (LCF/BQ/EU19/11710071) and Centro Superior de Investigaciones Científicas (JAEINT18/EX/0636).

References

1. Schumacher TN, Schreiber RD. Neoantigens in cancer immunotherapy. Science. 2015;348:69–74. doi:10.1126/science.aaa4971.

2. Waldman AD, Fritz JM, Lenardo MJ. A guide to cancer immunotherapy: From t cell basic science to clinical practice. Nature Reviews Immunology. 2020. doi:10.1038/s41577-020-0306-5.

3. Hundal J, Carreno BM, Petti AA, Linette GP, Griffith OL, Mardis ER, et al. pVAC-Seq: A genome-guided in silico approach to identifying tumor neoantigens. Genome Medicine. 2016;8:1–11.

4. Richters MM, Xia H, Campbell KM, Gillanders WE, Griffith OL, Griffith M. Best practices for bioinformatic characterization of neoantigens for clinical utility. Genome medicine. 2019;11:56.

5. Rubinsteyn A, Mondet S, Finnigan JP, Kodysh J, Hodes I, Hammerbacher J, et al. Computational Pipeline for the PGV-001 Neoantigen Vaccine Trial. Frontiers in Immunology. 2018;8 January:1–7. doi:10.3389/fimmu.2017.01807.

6. Kim S, Kim H, Kim E, Lee M, Shin E-C, Paik S, et al. Neopepsee: Accurate genome-level prediction of neoantigens by harnessing sequence and amino acid immunogenicity information. Annals of Oncology. 2018;29:1030–6. doi:https://doi.org/10.1093/annonc/mdy022.

7. Wang T-Y, Wang L, Alam SK, Hoeppner LH, Yang R. ScanNeo: identifying indel-derived neoantigens using RNA-Seq data. Bioinformatics. 2019;35:4159–61. doi:10.1093/bioinformatics/btz193.

8. Wood MA, Nguyen A, Struck AJ, Ellrott K, Nellore A, Thompson RF. neoepiscope improves neoepitope prediction with multivariant phasing. Bioinformatics. 2019;36:713–20. doi:10.1093/bioinformatics/btz653.

9. Bjerregaard AM, Nielsen M, Hadrup SR, Szallasi Z, Eklund AC. MuPeXI: prediction of neo-epitopes from tumor sequencing data. Cancer Immunology, Immunotherapy. 2017;66:1123–30.

10. Mösch A, Raffegerst S, Weis M, Schendel DJ, Frishman D. Machine learning for cancer immunotherapies based on epitope recognition by t cell receptors. Frontiers in Genetics. 2019;10:1141. doi:10.3389/fgene.2019.01141.

11. Duan F, Duitama J, Al Seesi S, Ayres CM, Corcelli SA, Pawashe AP, et al. Genomic and bioinformatic profiling of mutational neoepitopes reveals new rules to predict anticancer immunogenicity. Journal of Experimental Medicine. 2014;211:2231–48.

12. Bjerregaard A-M, Pedersen TK, Marquard AM, Hadrup SR. Prediction of neoepitopes from murine sequencing data. Cancer Immunology, Immunotherapy. 2019;68:159–61.

13. Lundegaard C, Lamberth K, Harndahl M, Buus S, Lund O, Nielsen M. NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8–11. Nucleic Acids Research. 2008;36 suppl_2:W509–12. doi:10.1093/nar/gkn202.

14. DeVette CI, Andreatta M, Bardet W, Cate SJ, Jurtz VI, Jackson KW, et al. NetH2pan: A Computational Tool to Guide MHC Peptide Prediction on Murine Tumors. Cancer Immunology Research. 2018;6:636–44. doi:10.1158/2326-6066.cir-17-0298.

15. Bhattacharya R, Sivakumar A, Tokheim C, Guthrie VB, Anagnostou V, Velculescu VE, et al. Evaluation of machine learning methods to predict peptide binding to MHC Class I proteins. bioRxiv preprint bioRxiv:154757. 2017. doi:10.1101/154757.

16. Lipton ZC, Berkowitz J, Elkan C. A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:150600019. 2015.

17. Sønderby SK, Winther O. Protein secondary structure prediction with long short term memory networks. arXiv preprint arXiv:14127828. 2014.

18. Hsieh Y-L, Chang Y-C, Chang N-W, Hsu W-L. Identifying protein-protein interactions in biomedical literature using recurrent neural networks with long short-term memory. In: Proceedings of the eighth international joint conference on natural language processing (volume 2: Short papers). 2017. pp. 240–5.

19. Xu C. A review of somatic single nucleotide variant calling algorithms for next-generation sequencing data. Computational and Structural Biotechnology Journal. 2018;16:15–24.

20. Andrews S. FastQC - A quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/. 2010;2019. doi:citeulike-article-id:11583827.

21. Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, et al. STAR: Ultrafast universal RNA-seq aligner. Bioinformatics. 2013;29:15–21. doi:10.1093/bioinformatics/bts635.

22. Broad Institute. Picard toolkit. Broad Institute, GitHub repository. 2019.

23. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, et al. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. Genome research. 2010;20:1297–303. doi:10.1101/gr.107524.110.

24. Van der Auwera GA, Carneiro MO, Hartl C, Poplin R, Angel G del, Levy-Moonshine A, et al. From fastQ data to high-confidence variant calls: The genome analysis toolkit best practices pipeline. Current Protocols in Bioinformatics. 2013;43:1–33.

25. Poplin R, Ruano-Rubio V, DePristo MA, Fennell TJ, Carneiro MO, Van der Auwera GA, et al. Scaling accurate genetic variant discovery to tens of thousands of samples. bioRxiv preprint bioRxiv:201178. 2018. doi:10.1101/201178.

26. Cirulli ET, Singh A, Shianna KV, Ge D, Smith JP, Maia JM, et al. Screening the human exome: A comparison of whole genome and whole transcriptome sequencing. Genome Biology. 2010;11.

27. Coudray A, Battenhouse AM, Bucher P, Iyer VR. Detection and benchmarking of somatic mutations in cancer genomes using rna-seq data. PeerJ. 2018;6:e5362.

28. McLaren W, Gil L, Hunt SE, Riat HS, Ritchie GRS, Thormann A, et al. The Ensembl Variant Effect Predictor. Genome biology. 2016;17:122. doi:10.1186/s13059-016-0974-4.

29. Bateman A, Martin MJ, O'Donovan C, Magrane M, Apweiler R, Alpi E, et al. UniProt: A hub for protein information. Nucleic Acids Research. 2015;43:D204–12.

30. Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, Baren MJ van, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nature biotechnology. 2010;28:511–5. doi:10.1038/nbt.1621.

31. Vita R, Dhanda SK, Martini S, Mahajan S, Sette A, Peters B, et al. The Immune Epitope Database (IEDB): 2018 update. Nucleic Acids Research. 2018;47:D339–43. doi:10.1093/nar/gky1006.

32. Smith T, Waterman M. Identification of common molecular subsequences. 1981;147:195–7. doi:10.1016/0022-2836(81)90087-5.

33. Henikoff S, Henikoff JG. Amino acid substitution matrices from protein blocks. Proceedings of the National Academy of Sciences of the United States of America. 1992;89:10915–9.

34. Chollet F, others. Keras. 2015.

35. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-scale machine learning on heterogeneous systems. 2015. http://tensorflow.org/.

36. Castle JC, Kreiter S, Diekmann J, Löwer M, Roemer N van de, Graaf J de, et al. Exploiting the mutanome for tumor vaccination. Cancer Research. 2012;72:1081–91. doi:10.1158/0008-5472.CAN-11-3722.

37. Devlin J, Chang M, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. CoRR. 2018;abs/1810.04805. http://arxiv.org/abs/1810.04805.

38. Jing L, Gülçehre, Peurifoy J, Shen Y, Tegmark M, Soljacic M, et al. Gated orthogonal recurrent units: On learning to forget. CoRR. 2017;abs/1706.02761. http://arxiv.org/abs/1706.02761.

39. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. 2014.

40. Vogelstein B, Papadopoulos N, Velculescu VE, Zhou S, Diaz LA, Kinzler KW. Cancer genome landscapes. Science. 2013;339:1546–58.

Figures

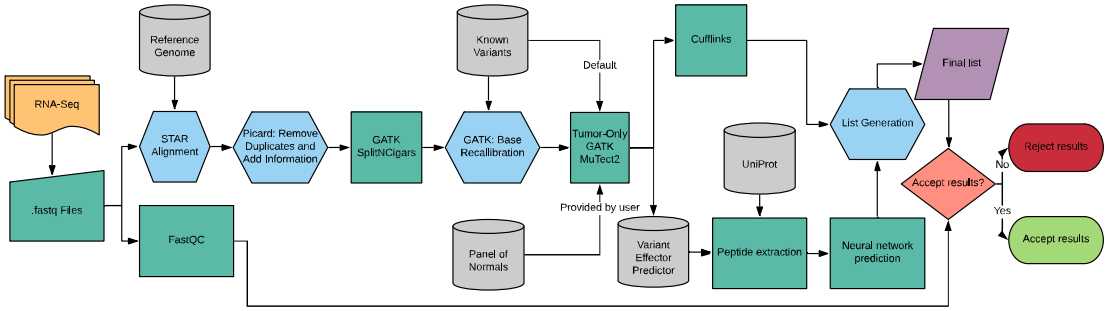

Figure 1 Workflow for the integrated pipeline. Firstly, the sample is preprocessed before variant calling. Quality control through FastQC and STAR alignment with the reference genome is followed with protocols from Best Practices of GATK. Known variants are introduced through known polymorphisms or a panel-of-normals, if it is requested and a sufficient non-tumoral RNA-Seq reads are provided. MuTect2 is used for variant calling and plausible single nucleotide variant (SNV) mutations are translated into peptidic sequences to be used for prediction with the RNN model. Gene expression is quantified through Cuffquant in Cufflinks.

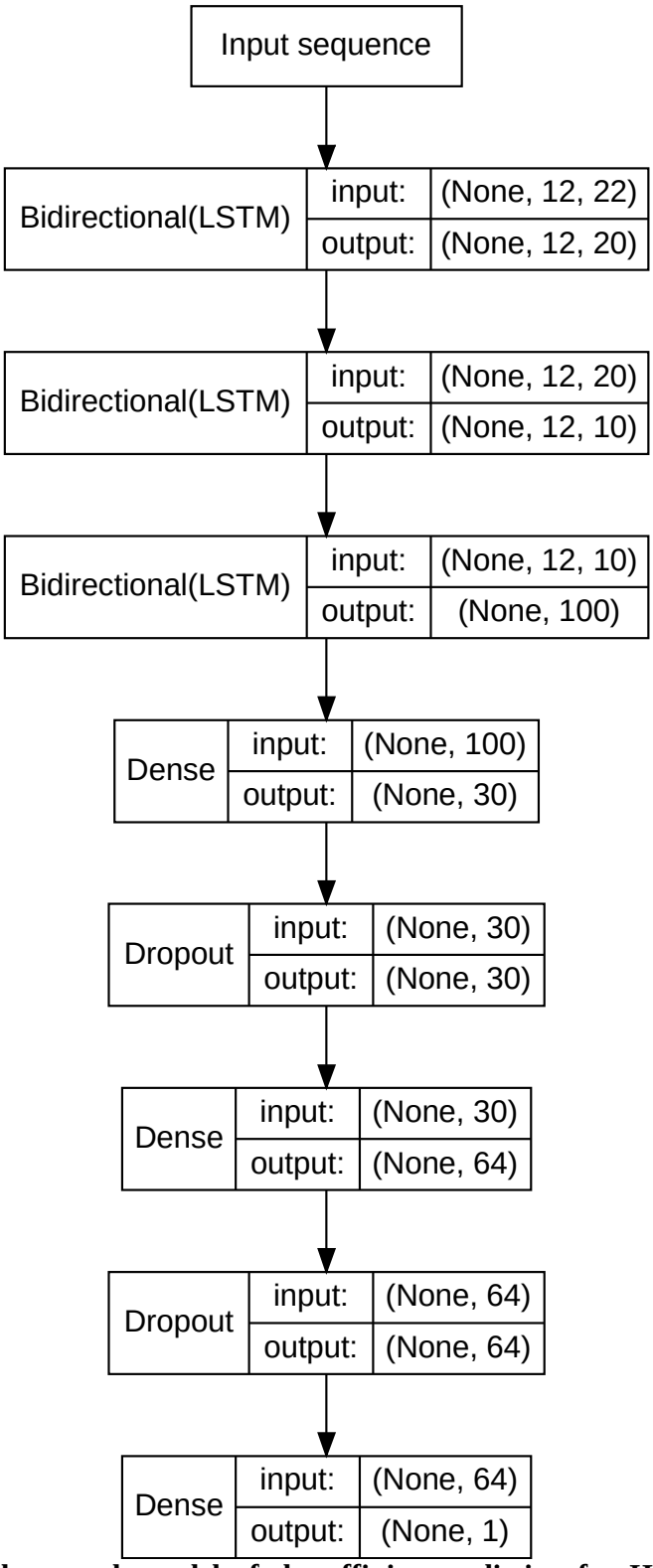

Figure 2 Neural network model of the affinity prediction for H-2K^d. The input sequence corresponds to a one-hot encoding of a 12mer peptide sequence extracted from the preprocessing workflow. The number of LSTM units corresponds to the overall length of the input sequence across the three consecutive layers. Following the

RNN, two hidden dense units, with alternating dropouts, serve to process an affinity probability.

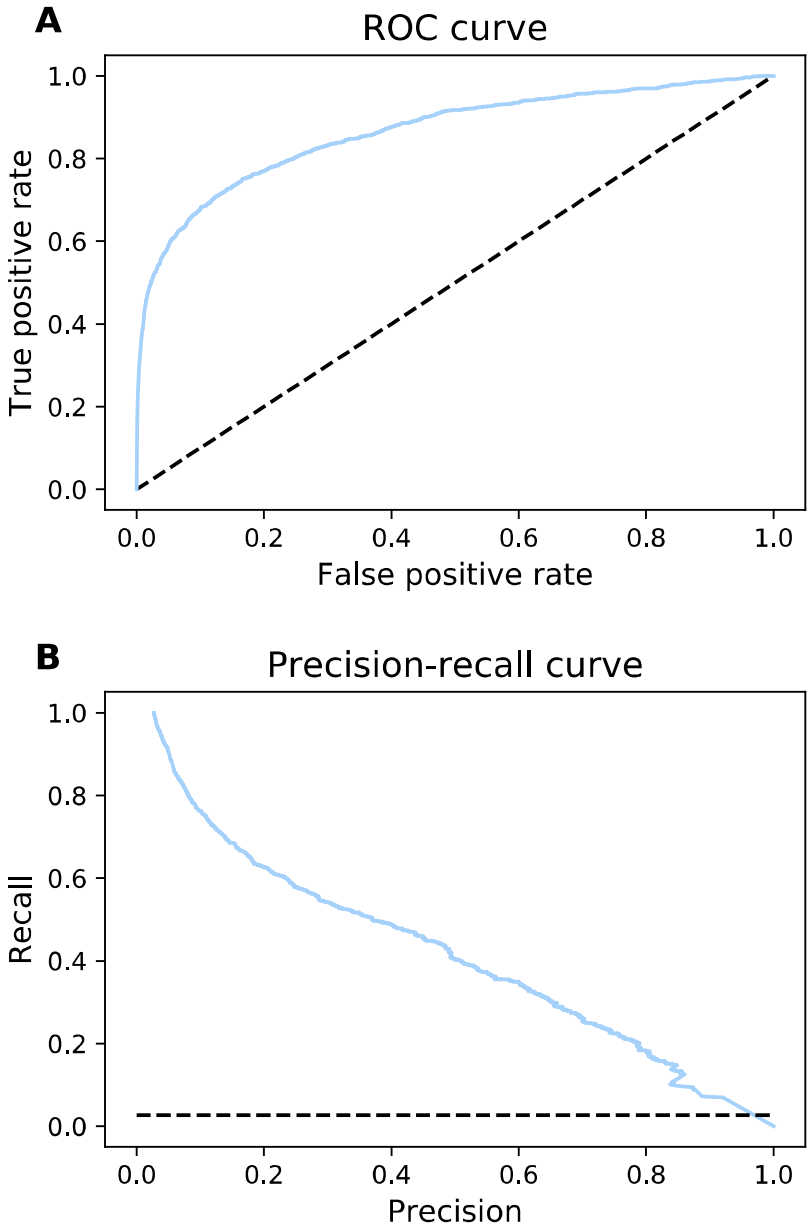

Figure 3 ROC and precision-recall curves for final model. A) ROC curve for 10% test partition with an AUC of 86.5%, the dashed line shows chance level. **B)** Precision-recall curve with the prevalence of around 3% shown as chance. The AUC of the precision-recall is of 41.97% whereas a random guess corresponds to an AUC of 2.64% for the same data imbalance.

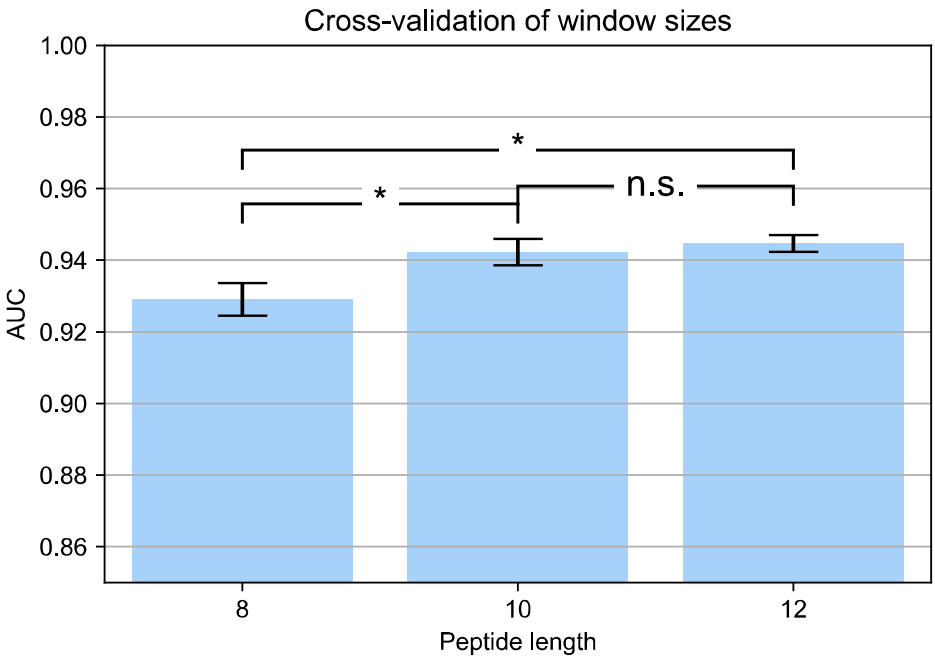

Figure 4 Cross-validation of peptide window sizes. Area under the curve of the reciever operating characteristic curve using 8mers, 9mers and 12mers obtained through 5-fold cross-validation in different conditions. The windows are obtained from the mutated peptide sequence centered at the location of the SNV. Significant differences between means ($p<0.05$) are shown.

Tables

Table 1 *Binary classification metrics for the final 5-fold cross-validated algorithm. The reported mean statistics estimators correspond to AUC ROC, accuracy (ACC), precision or positive predictive value (PPV) and sensitivity and specificity with their harmonic average (F1). Prevalence of positive sample was around 1:40.*

AUC ROC	ACC	PPV	Sensitivity	Specificity	F1
(±SD)	(±SD)	(±SD)	(±SD)	(±SD)	(±SD)
0.95 ± 0.04	0.977 ± 0.004	0.6 ± 0.1	0.62 ± 0.09	0.988 ± 0.004	0.6 ± 0.1

Table 2 *Putative neoantigens, shown by sequence and gene symbol, for the B16 melanoma model, restricted to H-2K^b, ranked by scores. The gene expression is quantified as fragments per kilobase million. Neoantigens examined in Castle et al. [36] are classified by selection for validation (*) and reactivity (**). The classification of the average score for a complete 12mer sequence given by NetH2pan is also presented.*

#	Sequence	Gene	Score	FPKM	Castle *et al.*	NetH2pan
1	NKVVMEYENLEK	Pnp	1.00	3.04	-	24
2	KASGFRYNVLSC	Nr1h2	1.00	0.00	-	1
3	SQAWTHPPGVVN	Adar	1.00	0.00	-	88
4	TFVYPTIFPLRE	Lrrc28	1.00	0.94	-	10
5	DKSYTLPSSLRK	Zic2	1.00	1.83	-	27
6	TLAQLTWPLWLE	Hjurp	0.43	0.00	-	26
7	VDTNMMGHEHIR	Safb2	0.26	24.20	-	140
8	AKTAVNDYFQCN	Stox2	0.25	0.00	-	126
9	FIAIYHHASRAI	Tm9sf3	0.21	24.29	**	8
10	SGASNTTPHLGF	Tab2	0.20	29.21	-	103
11	YSSMRMMKEALQ	Herc6	0.18	10.93	-	38
12	TRASVTNFQIVH	Tulp2	0.16	0.00	-	43
13	AWGVDGTLAQLE	Pkdcc	0.16	5.50	-	118
14	VVLLMDALYLLR	Sirpa	0.14	51.24	-	13
15	NVTISNLYEGMM	Hjurp	0.13	0.00	-	6
16	ARALWFWAFSLQ	Sfi1	0.09	0.00	-	5
17	GASSFREAMRIG	Eno3	0.09	29.01	-	21
18	LAAIVGKQVLLG	Rpl13a	0.09	1203.49	*	67
19	AYSAHTSENLED	Zfp638	0.09	0.00	-	142
20	TVAVLGFILSSA	Commd4	0.09	41.28	-	52
21	FQYCLFKICRDV	Pla2g12a	0.08	7.05	-	63
22	AISAPCIGSPGC	Hjurp	0.08	0.00	-	227
23	HKHLMPTQIIPG	Jmjd1c	0.08	3.42	-	144
24	MFGIDGFAAVIN	Pdhx	0.07	10.26	-	56
25	YQPRQSVSYEDV	Tasor2	0.06	5.16	-	188
26	LCPLESRVPHTL	Hjurp	0.06	0.00	-	218

27	QMIVFYLIELLK	Jak2	0.05	6.03	-	2
28	AHMYEAVALIKD	Dennd5a	0.05	64.21	-	17
29	DRIVHALNTTVP	Ccdc58	0.05	0.00	-	70
30	NEVDVQEVTHSA	Dlg4	0.04	9.45	-	289
31	LAAIVGKQVLLV	Rpl13a	0.04	1203.49	*	48
32	QRNRKLDYSSSE	Bod1l	0.04	3.65	-	282
33	HLGCIKKKFLQR	Sfi1	0.04	0.00	-	177
34	PPTARMMFSGLA	Wiz	0.03	16.70	-	18
35	QEEVFAKHVSNA	Smarcc2	0.03	0.00	-	167

Additional Files

Additional file 1 — Supplementary material
Additional file 2 — Grid search parametrization