Identification of Epitope-TCR Binding Using A Generative Adversarial Network Model

Seojin Bang [1] **Heewook Lee** [2]

Abstract

Testing immunogenicity of an epitope is important as epitope-TCR binding is one of the fundamental mechanisms of adaptive immune system. Previously developed methods use discriminative models to answer whether a given epitope-TCR pair binds or not. However, they are intrinsically limited because combinatorially many epitope-TCR pairs exist, making it difficult to effectively screen for highly immunogenic pairs. Here, we propose to use a generative model based on GAN to output a distribution of TCR sequences given an epitope. This approach is more effective in practice as effective solutions for finding target epitopes exist, especially for cancer immunotherapy. We show that our model can produce TCR sequences that plausibly bind to a given epitope. We also show that our model can be used as a tool for predicting epitope-TCR binding affinity with better prediction performance.

1. Introduction

Epitope-TCR binding is a key mechanism of the adaptive immune system. An epitope is a part of an antigen that can trigger immune responses by binding to cognate T cell receptors (TCR), a molecule on the surface of a T cell. Therefore, assessing the immunogenicity of an epitope is an important problem as screening for a pair of epitope and its cognate TCRs is necessary for clinical applications such as cancer immunotherapy (Mösch et al., 2019) or treatment of autoimmune conditions.

An individual person harbors a large number of distinct TCRs, approximately 10^{10} (Lythe et al., 2016) because each T cell creates its own distinct TCR sequence by recombinations of multiple gene segments and random mutations during T cell maturation. Having a huge number of different TCRs (TCR repertoire) allows the host to be adaptive to largely varying antigen sequences generated by invaders such as viruses and pathogens.

In cancer immunotherapy, researchers need to find (1) candidate neoantigens patient's tumor specific antigen, and (2) either find cognate TCR or engineer T cells to recognize the neoepitopes. To facilitate the second task, being able to computationally identify which TCR will bind to a specific epitope is essential. However, this is a challenging task due to a large number of TCRs and epitopes. It is further exacerbated by the fact that single TCR can interact with many epitopes and that an epitope can also interact with a large number of TCRs, making tremendously large combinations of epitope-TCR pairs.

Existing approaches have focused on discriminating whether a given epitope-TCR pair will bind or not. NetTCR (Jurtz et al., 2018) predicts binding affinity of given epitope-TCR pairs, and TCRGP (Jokinen et al., 2019) and TCRex (Gielis et al., 2019) build an epitope-specific model that predicts binding affinity of given TCR. These approaches rely on known interacting epitope-TCR pairs available from public databases such as VDJdb (Shugay et al., 2017) and IEDB (Vita et al., 2014). However, these approaches are inherently limited because such a framework requires an enumeration of a massive number of epitope-TCR pairs, and they can only test sequences that are known.

To overcome such shortcomings of previously developed methods, we propose generating a class of TCRs that can bind to a given epitope sequence. This way, we can easily screen within a patient's TCR repertoire as well as engineer T cells based on the generated TCR sequences.

A generative model learns the probability distribution of data and enables to produce realistic samples based on that (Goodfellow, 2016). In our task, a single epitope corresponds to many different TCR sequences. When we use traditional machine learning models by minimizing the mean squared error between a desired and predicted TCR sequences, we cannot produce multiple different TCR se-

[1]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA [2]School for Computing, Informatics, and Decision Sciences Engineering and Biodesign Institute, Arizona State University, Tempe, AZ, USA. Correspondence to: Seojin Bang <seojinb@cs.cmu.edu>, Heewook Lee <heewook.lee@asu.edu>.

Presented at the ICML 2020 Workshop on Computational Biology (WCB).

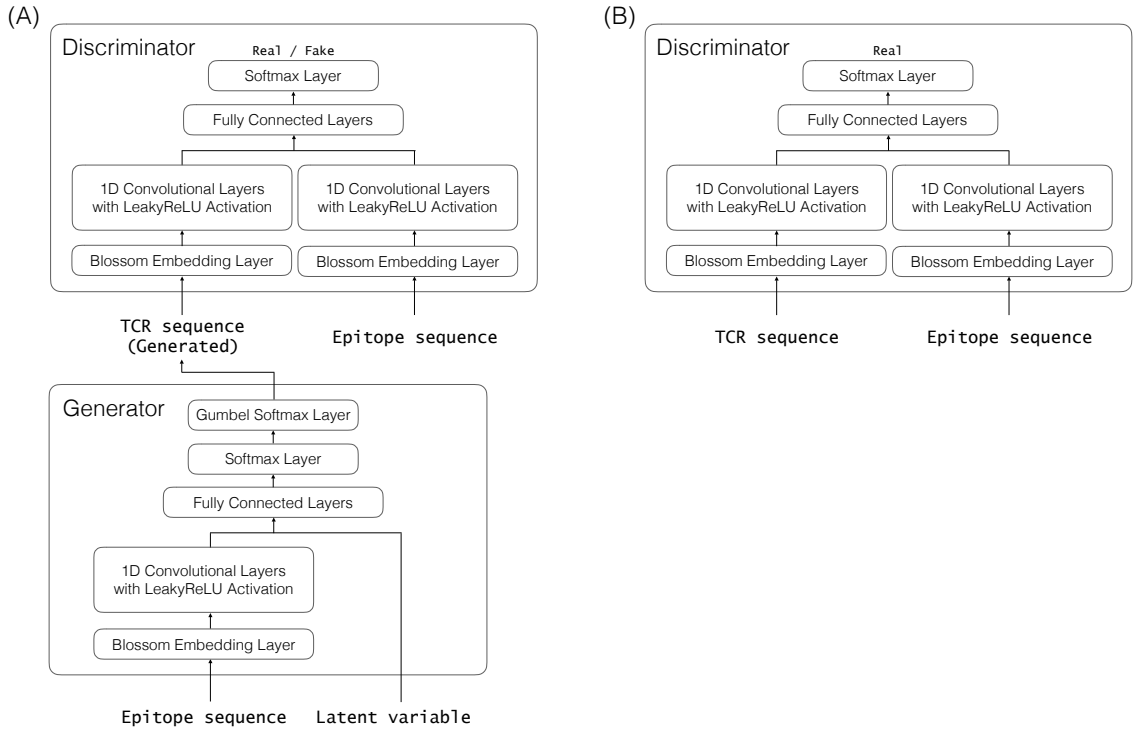

Figure 1. Generative model for synthesizing TCR sequences bind to given epitopes, regarding to (A) fake TCR and epitope pairs, and (B) real TCR and epitope pairs.

quences. Generative models such as GANs (Goodfellow et al., 2014) enables us to learn multi-modal distribution, which is desired for generating a class of TCRs binding to a given epitope.

We develop an approach using a generative model for synthesizing TCR sequences that plausibly bind to given epitopes based on GAN. Our model can be seen as a novel application of conditional GAN (cGAN, Mirza & Osindero (2014)) to a biological sequence generation problem. cGAN is a type of GAN constructed by feeding class labels of data to both the generator and discriminator, which enables us to generate multi-modal distribution conditioned on each label. Our task is to generate TCR equivalents of an epitope that resembles real TCR sequences. In our model, epitopes work as feeding class labels of data we wish to condition on, and TCRs are the generated samples conditioned on the class labels. It allows us to generate a multi-modal TCR distribution conditioned on each epitope.

2. Method

2.1. Generative Model

Our model is composed of a **generator** and **discriminator** learned in an adversarial manner (Figure 1). The generator G creates TCR sequences $\mathbf{t}^*$ (asterisk indicates that the sequence is synthesized) that resemble real TCR sequences matching to an epitope $\mathbf{e}$. The discriminator D takes a pair of TCR and epitope sequences, $(\mathbf{e}, \mathbf{t})$ or $(\mathbf{e}, \mathbf{t}^*)$, as an input and determines whether the TCR sequence is real or fake about the given epitope sequence.

In the training process, they are adversarially learned in a minimax framework. The discriminator is learned to distinguish real and fake pairs. It minimizes a binary cross-entropy loss defined over real and fake pairs; real pairs where both TCR and epitope are from the dataset are labeled as 1, and fake pairs where TCR sequences are synthesized are labeled as 0:

$$\mathcal{L}(D \mid G) = -\frac{1}{2}\mathbb{E}_{(\mathbf{e},\mathbf{t})\sim\mathbb{P}(\mathbf{e},\mathbf{t})} log D(\mathbf{e}, \mathbf{t}) - \frac{1}{2}\mathbb{E}_{\mathbf{e}\sim\mathbb{P}(\mathbf{e}),\mathbf{z}\sim\mathbb{P}(\mathbf{z})} log(1 - D(\mathbf{e}, \mathbf{t}^*))$$

where $\mathbf{t}^* = G(\mathbf{e}, \mathbf{z})$. The generator is learned to deceive the discriminator. It minimizes a *fake loss* that are defined over the synthesized TCR and epitope pairs by flipping the label (a binary cross-entropy loss of fake pairs labeled as 1):

$$\mathcal{L}(G \mid D) = -\frac{1}{2}\mathbb{E}_{\mathbf{e}\sim\mathbb{P}(\mathbf{e}),\mathbf{z}\sim\mathbb{P}(\mathbf{z})} log D(\mathbf{e}, \mathbf{t}^*)$$

which is equivalent to maximizing the log-likelihood of the discriminator fails to detect fake pairs.

2.2. Reparameterization Trick

The generator returns multiple channels of softmax layer corresponding to each position in TCR sequence; for each position j, the softmax layer of generator returns probabilities for 24 amino acids, said $(\mathbf{p}_{j,1}, \cdots, \mathbf{p}_{j,24})$. During inference, we simply take an argmax within each softmax layer and obtain the most feasible amino acid at each position. However, during training, the argmax layer makes the gradient computing impracticable as it is not differentiable. In order to address this problem, we use a Gumbel-softmax layer to make it differentiable (Jang et al., 2017). It is a differentiable approximation to a non-differentiable categorical sampling using reparameterization, enabling us to use standard backpropagation to compute gradients. The argmax layer is necessary in learning discriminator. If we simply use the probabilities obtained from the softmax layer as an input to the discriminator, a real TCR having a form of discrete sequences cannot be used as an input in the learning. In the same manner, it also enables to use the discriminator as a tool for predicting epitope-TCR binding affinity as illustrated in Section 4.2.

For each position j, we independently sample random variables $\mathbf{r}_{j,1}, \cdots, \mathbf{r}_{j,24}$ corresponding to 24 amino acids and obtain a Concrete random vector defined as follows:

$$\mathbf{c}_{j,a} = \frac{\exp\left(\left(-\log\left(-\log \mathbf{r}_{j,1}\right) + \log \mathbf{p}_{j,a}\right)/\tau\right)}{\sum_{a=1}^{24} \exp\left(\left(-\log\left(-\log \mathbf{r}_{j,1}\right) + \log \mathbf{p}_{j,a}\right)/\tau\right)}$$
$$\text{where } \mathbf{r}_{j,a} \sim U(0,1)$$

where τ is a tuning parameter that controls uniformness of the random vector $(\mathbf{c}_{j,1}, \cdots, \mathbf{c}_{j,24})$. The random vector serves as a continuous, differentiable approximation to the argmax layer for each position j.

3. Experiment

3.1. Data and Preprocessing

We use the two largest public databases of known interacting epitope-TCR pairs: VDJdb (Shugay et al., 2017) and IEDB (Vita et al., 2014). VDJdb dataset contains 2,892 epitope-TCR pairs and IEDB dataset contains 9,328 epitope-TCR pairs. We use the samples pre-screened by Jokinen et al. (2019) and Jurtz et al. (2018), respectively.

Figure 2 illustrates data and preprocessing pipeline of our analysis. We randomly split the VDJdb samples into training, validation, and test sets, which include 80%, 10%, and 10% of data, respectively. It is composed of 21 unique epitope sequences with 138 TCR sequences per epitope in average. We use the IEDB samples as independent test set for evaluating out-of-sample performance of our approach. There are 6 epitopes contained in both VDJdb and IEDB: GILGFVFTL, GLCTLVAML, NLVPMVATV, LLWNGPMAV, YVLDHLIVV, CINGVCWTV. Each TCR and

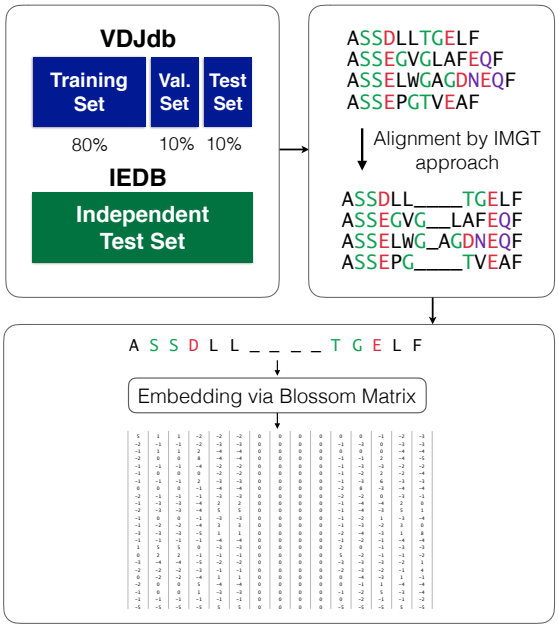

Figure 2. Data and preprocessing pipeline.

epitope sequence is aligned according to the IMGT definitions (Lefranc et al., 2005); we pad the sequences with 0 in the middle of the sequence. The BLOSUM50 matrix (Henikoff & Henikoff, 1992) is used for initializing the embedding matrix because it provides a prior knowledge about which amino acids are biologically similar to each other.

3.2. Experimental Detail

The generator consists of one encoder processing epitope sequence, two fully connected layers concatenating epitope and latent variables, a softmax layer, and a Gumbel-softmax layer for the reparameterization trick. The encoder consists of a dropout layer with the rate 0.3, three convolutional layers with the kernel size 3 followed by a batch normalization layer, and a Leaky-ReLU activation function, and one 1D convolutional layer followed by a hyperbolic tangent function. The discriminator consists of two encoders processing each TCR and epitope sequence, two fully connected layers concatenating TCR and epitope sequences, and a softmax layer determining whether they are a real or fake pair binds together. Each encoder consists of two convolutional layers with the kernel size 3 followed by a batch normalization layer, a Leaky-ReLU activation function, and a dropout layer with the rate 0.3.

We optimize the model with the following search space (bold indicate the choice for our final model): the batch size – $\{\mathbf{50}, 100\}$, learning rate for generator – $\{0.05, 0.01, \mathbf{0.001}, 0.0001\}$, learning rate for discriminator – $\{0.05, 0.01, \mathbf{0.001}, 0.0001\}$, the maximum length for TCR – $\{12, \mathbf{15}, 20\}$, the maximum length of epi-

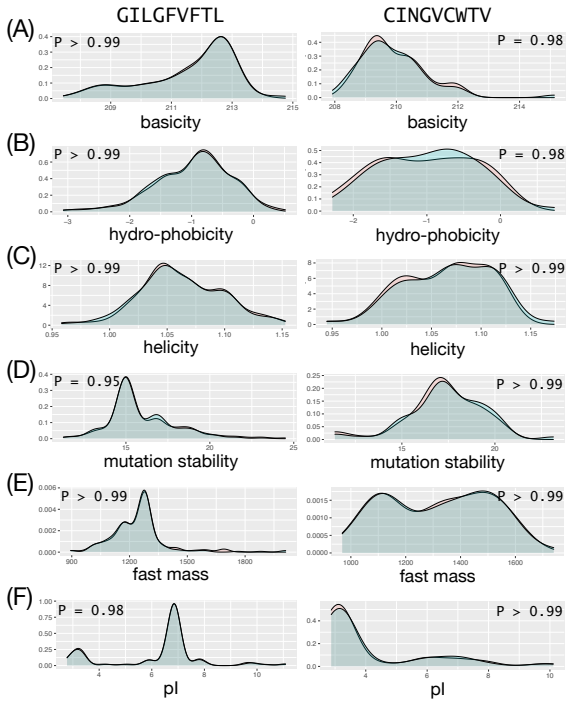

Figure 3. Comparison of physiochemical properties of synthesized and true TCR distributions. The results of TCR distributions corresponding to the two epitopes (*GILGFVFTL* and *CINGVCWTV*) are shown. The properties of a TCR sequence is calculated by averaging the amino acid properties in the sequence. P-values from Kolmogorov-Smirnov Test are shown where the larger p-value supports similarity between the two distribution.

tope – $\{8, \mathbf{13}\}$, and the dimension of latent variable – $\{1, 10, 50, \mathbf{100}\}$. They are tuned via grid search and learned via Adam algorithm (Kingma & Ba, 2014).

4. Result

4.1. Synthesized TCR in comparison to real TCR

We evaluate the generated TCR sequences based on how similar they are to the true distribution. In order to compare the synthesized and true distribution, we first quantify each TCR sequence using the following physiochemical properties as described in Degroeve & Martens (2013): *basicity*, *hydrophobicity*, *helicity*, *mutation stability*, *fass mass*, and *pI*. We then compare distribution of the physiochemical properties between the synthesized and true TCR groups. We statistically assess the similarity between the two distributions using Two-Sample Kolmogorov-Smirnov Test (Conover & Conover, 1980) where the larger p-value supports similarity between the two distribution. Figure 3 shows the distributions of physiochemical properties between synthesized and true TCRs are significantly similar to each other. We visually represent the synthesized TCRs in Figure 4, which also supports resemblance of the two

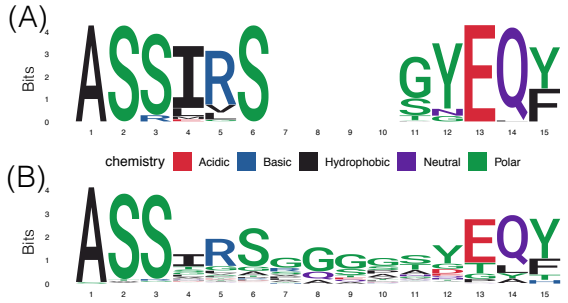

Figure 4. Visualization of (A) synthesized and (B) true distribution of TCRs matching to epitope *GILGFVFTL*.

Table 1. Prediction performance on epitope-TCR binding affinity

	VDJdb Test Set		IEDB Indep. Set
	Recall	Precision	Recall
Our method	84.08	70.03	60.30
CNN baseline	76.21	70.83	44.39

distributions.

4.2. Discriminator as a tool for assessing epitope-TCR binding affinity

We examine whether the discriminator can be used as a tool for assessing epitope-TCR binding affinity. In our model, the discriminator returns log-probability representing a reliability score about whether the TCR sequence is real or fake about a given epitope. We examine whether this can be used as a reliability score about the binding affinity of epitope-TCR pairs, as other existing models such as NetTCR and TCRGP provide. We evaluate prediction performance of our model in three ways: (1) the recall on VDJdb test set composed of 2,892 positively binding epitope-TCR pairs, (2) the precision on VDJdb test set composed of 2,892 positively binding pairs and 2,892 synthesized negatively binding pairs provided by Jokinen et al. (2019), and (3) the out-of-sample prediction performance (recall) on IDEB independent test set composed of positive-bind epitope-TCR pairs. The baseline method is a deep neural network using convolution layers predicting the binding affinity of given epitope-TCR pairs, as NetTCR does. It takes an epitope-TCR pair as an input and returns log-probability score representing binding affinity between the two. As seen in Table 1, our method outperforms the baseline method in prediction of the positive binds. It especially works better in the out-of-sample prediction problem. We note that our method does not use the synthesized negative pairs in learning the model. However, our method shows a comparable performance with the baseline that uses the negative pairs in learning the model. Those results support that our discriminator can be used as a reliability score about the binding affinity.

References

Conover, W. J. and Conover, W. J. *Practical nonparametric statistics*. Wiley New York, 1980.

Degroeve, S. and Martens, L. Ms2pip: a tool for ms/ms peak intensity prediction. *Bioinformatics*, 29(24):3199–3203, 2013.

Gielis, S., Moris, P., Bittremieux, W., De Neuter, N., Ogunjimi, B., Laukens, K., and Meysman, P. Detection of enriched T cell epitope specificity in full T cell receptor sequence repertoires. *Frontiers in Immunology*, 10:2820, 2019.

Goodfellow, I. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Henikoff, S. and Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *International Conference on Learning Representations*, 2017.

Jokinen, E., Heinonen, M., Huuhtanen, J., Mustjoki, S., and Lähdesmäki, H. Tcrgp: Determining epitope specificity of t cell receptors. *bioRxiv*, pp. 542332, 2019.

Jurtz, V. I., Jessen, L. E., Bentzen, A. K., Jespersen, M. C., Mahajan, S., Vita, R., Jensen, K. K., Marcatili, P., Hadrup, S. R., Peters, B., et al. Nettcr: sequence-based prediction of tcr binding to peptide-mhc complexes using convolutional neural networks. *bioRxiv*, pp. 433706, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lefranc, M.-P., Pommié, C., Kaas, Q., Duprat, E., Bosc, N., Guiraudou, D., Jean, C., Ruiz, M., Da Piédade, I., Rouard, M., et al. Imgt unique numbering for immunoglobulin and t cell receptor constant domains and ig superfamily c-like domains. *Developmental & Comparative Immunology*, 29(3):185–203, 2005.

Lythe, G., Callard, R. E., Hoare, R. L., and Molina-París, C. How many TCR clonotypes does a body maintain? *Journal of Theoretical Biology*, 389:214–224, 2016.

Mirza, M. and Osindero, S. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

Mösch, A., Raffegerst, S., Weis, M., Schendel, D. J., and Frishman, D. Machine learning for cancer immunotherapies based on epitope recognition by T cell receptors. *Frontiers in Genetics*, 10, 2019.

Shugay, M., Bagaev, D. V., Zvyagin, I. V., Vroomans, R. M., Crawford, J. C., Dolton, G., Komech, E. A., Sycheva, A. L., Koneva, A. E., Egorov, E. S., et al. Vdjdb: a curated database of t-cell receptor sequences with known antigen specificity. *Nucleic Acids Research*, 46(D1):D419–D427, 2017.

Vita, R., Overton, J. A., Greenbaum, J. A., Ponomarenko, J., Clark, J. D., Cantrell, J. R., Wheeler, D. K., Gabbard, J. L., Hix, D., Sette, A., et al. The immune epitope database (iedb) 3.0. *Nucleic Acids Research*, 43(D1): D405–D412, 2014.