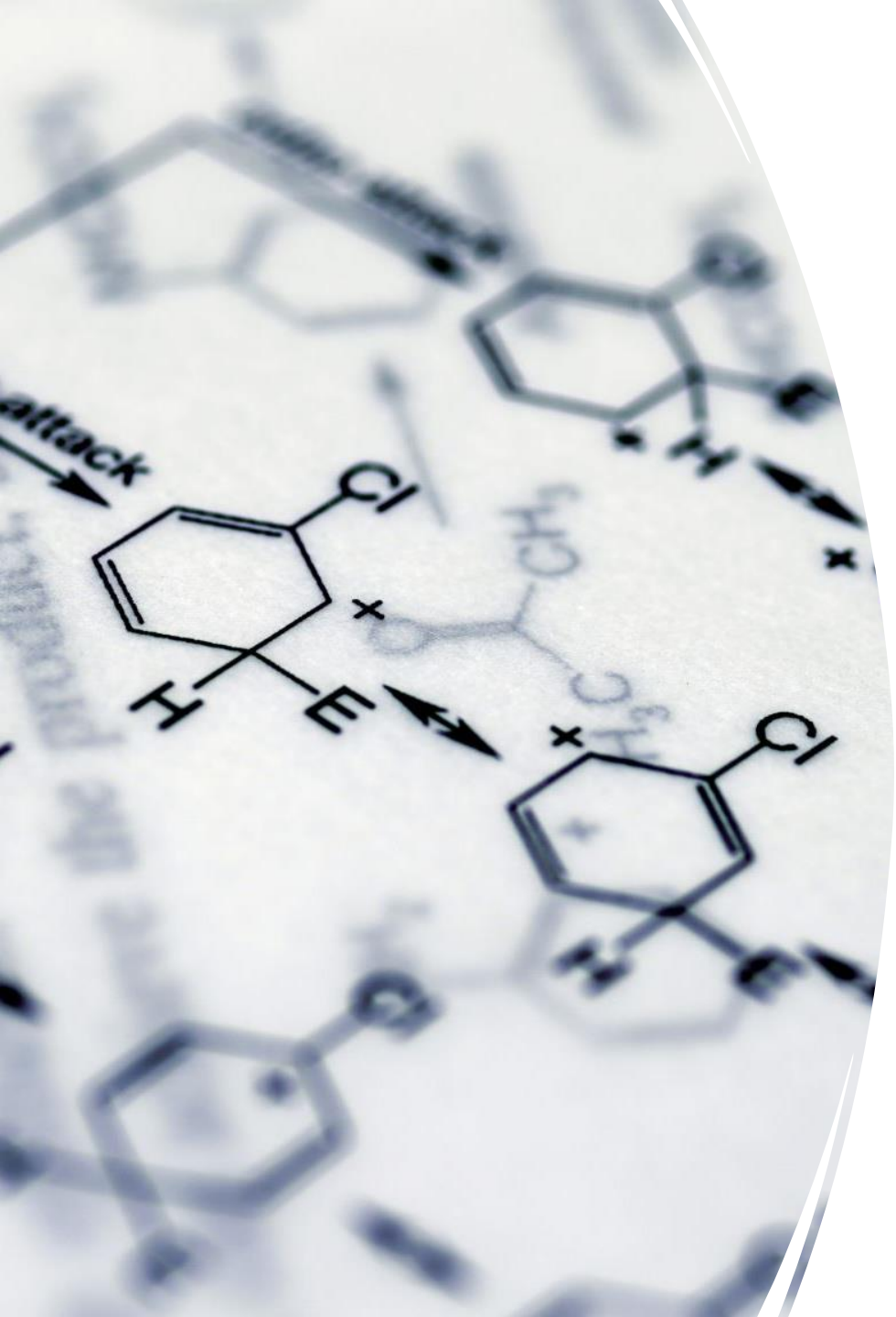

Antigen-Specific Antibody Design and Optimization with Diffusion-Based Generative Models for Protein Structures

Ismail Ceyhan

Advanced AI in Biomedicine – Winter Semester 2024

Table of Contents

Introduction

- Antibodies' role in the immune system
- Complementarity-determining regions (CDRs)

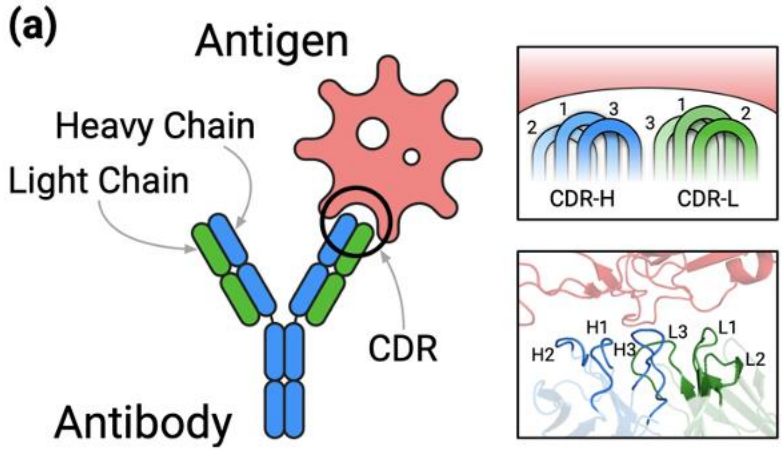

Figure 1 Antibody-antigen complex structure and CDR structure.

Background - I

- Traditional Computational Approaches
  - Rely on sampling algorithms over hand-crafted and statistical energy functions
  - Inefficicent and stuck in local optima
  - Not effectively model relationship between CDR sequences and their 3D structure.

Background - II

- Generative Models for Antibody Design
  - Recent models such as those proposed by Saka et al. (2021), Akbar et al. (2022) and Jin et al. (2022) focus on generating antibody sequences.
  - These Models use language models or other generative techniques to produce new antibody sequences.
  - Do not generate antibodies specifically tailored to the 3D structures of antigens.
  - Lack of ability to model complex interactions.

Background - III

- Jin et al.'s CDR Sequence-Structure Co-Design Model
  - This model focuses on designing antibodies to neutralize specific pathogens like SARS-CoV-2.
  - The model is not generalizable to arbitrary antigens.
  - It does not consider the orientation of amino acids.

Problem Definition

- The proposed method jointly models the distribution of CDR sequences and their corresponding 3D structures.
- Challenges:
  - large search space
  - need for high specificity
  - structural precision
  - generalizable approach

Proposed Solution - I

The diffusion-based generative model is introduced with following capabilities:

Joint Sampling of CDR Sequences and Structures

- The model generates CDRs by considering the 3D structure of antigen.

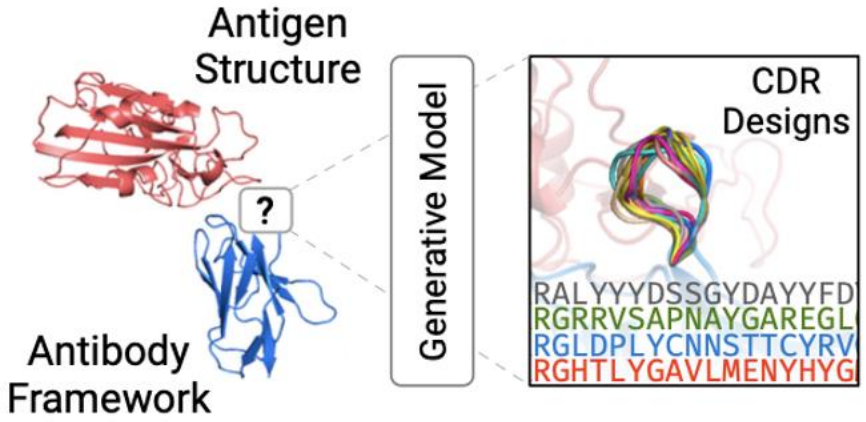

Figure 2 The task in this work is to design CDRs for a given antigen structure and an antibody framework.

Proposed Solution - II

Atomic-Level Design with Side-Chain Packing

- The model predicts side-chain orientations to achieve atomic-resolution accuracy.

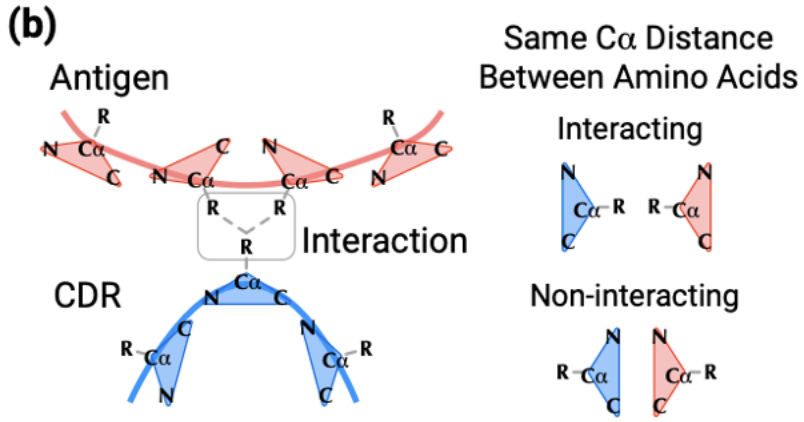

Figure 3 The orientations of amino acids (represented by triangles) determine their side-chain orientations, which are key to inter-amino-acid interactions.

Proposed Solution - III

Iterative Refinement Process

- The model employs an iterative update process that allows for continuous refinement of amino acid types, positions and orientations.
- The sequence-structure space more efficiently
- Avoiding to become trapped in local optima

Proposed Solution - IV

Flexible and Customizable Design Process

- The diffusion model allows iterative updates, enabling constraints and customizations during design process.
- This approach can be used in various tasks:
  - Sequence-structure co-design
  - Fix-backbone CDR design
  - Optimization of existing antibodies to enhance binding affinity.

Method – I – Model Framework

- The model designed to jointly sample CDR sequences and their corresponding 3D structures.
- Key components;
  - Diffusion Probabilistic Models
  - Equivarant Neural Networks

Method – II – Diffusion Probabilistic Model

Two Markov chains in the process;

- The forward diffusion process
  - It gradually adds noise to the data, transforming it into a simpler distribution. The process goes from *t=0* to *t=T*.
  - *The time t=0* represents the observed sequences and structures of CDRs and *t=T* represents samples from the prior distribution.
- The generative (backward) diffusion process
  - It starts from the simple distribution and iteratively refines it to generate samples from the target distribution. The generative diffusion goes backward from *t=T* to *t=0*.

Method – III – Input Data

- Protein Complex
  - An antigen and an antibody framework
- Initial CDR Configuraiton
  - The model starts with an arbitrary sequence of CDRs, including their positions and orientations.

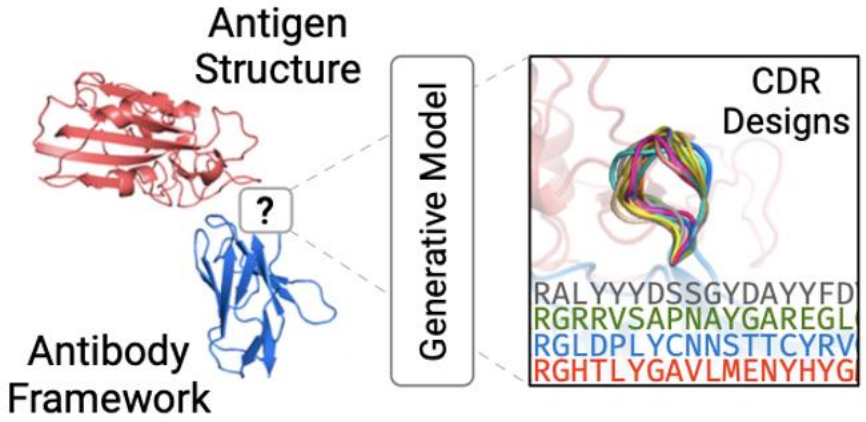

Figure 2 The task in this work is to design CDRs for a given antigen structure and an antibody framework.

Method – IV – Definitions and Notations

- Amino acids in a protein are represented by;
  - type
  - position of the Cα atom
  - orientation
- The type is denoted as s_i.
- The possible types are drawn from the set;

$$\{A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y\}$$

Method – V – Definitions and Notations

- The position of amino acid represented by the coordinate of its Cα atom, denoted as x_i.
- The orientation is described by O_i, which is a rotation matrix from the *SO(3)*.
- The type is denoted as s_i.
- The *antibody-antigen complex* is described by the set;

$$C = \{(s_i, x_i, O_i) \mid i \in \{1, \ldots, N\} \setminus \{l+1, \ldots, l+m\}\}$$

Method – VI – Definitions and Notations

- The CDR is generated consists of *m* amino acids, with indices ranging from *l+1* to *l+m*
  - where *l* is the starting index of the CDR
- *Each amino acid in the CDR* is represented by a set of components;

$$R = \{(s_j, x_j, O_j) \mid j = l + 1, \ldots, l + m\}$$

Method – VII – Initialization

- The process begins with a random sequence. This includes;
  - Amino Acid Sequence: A random selection of amino acids is made to form the initial CDR sequence.
  - Positions and Orientations: The model assigns initial spatial positions and orientations to the amino acids in the CDR.

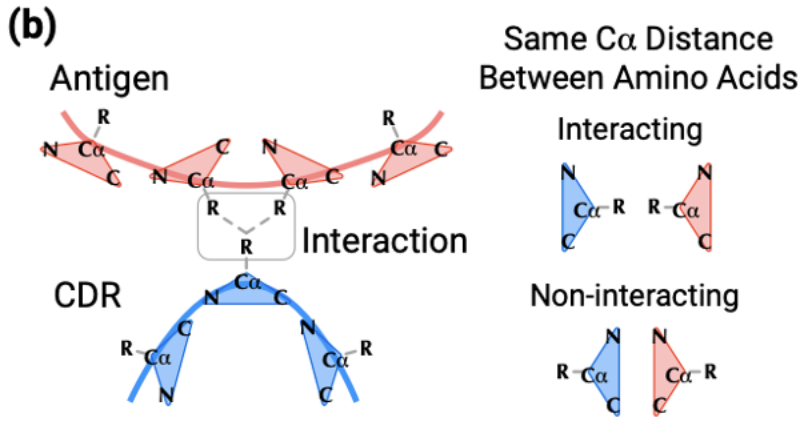

Figure 3 The orientations of amino acids (represented by triangles) determine their side-chain orientations, which are key to inter-amino-acid interactions.

Method – VIII – Iterative Update Process

- Information Aggregation
- Iterative Refinement
  - The model iteratively updates the parameters for each amino acid in the CDR.
    - Amino Acid Type: The model predicts the most suitable amino acid for each position
    - Position: The position of each amino acid is adjusted to optimize the fit
    - Orientation: The orientation of the side chains of amino acids is also updated

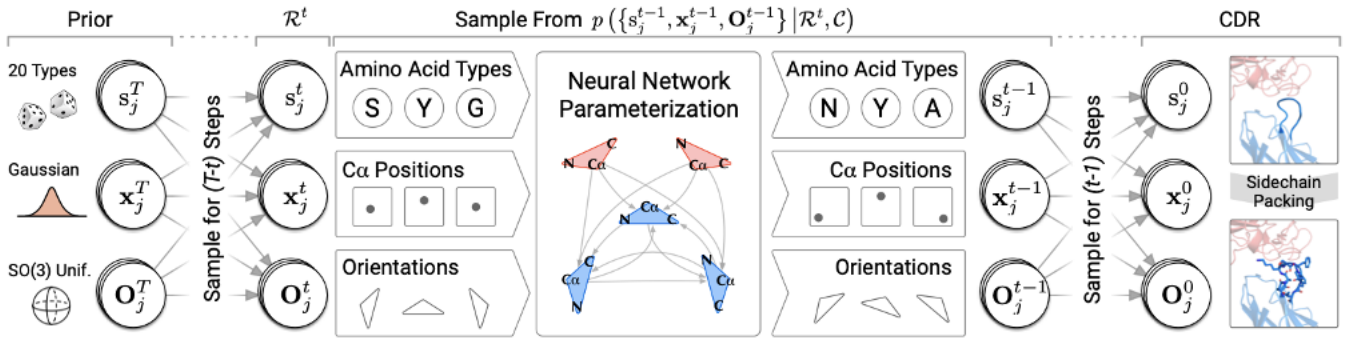

Figure 4 Illustration of the generative diffusion process.

Method – IX – Diffusion Process

- *Multimodel Diffusion for Amino Acid Types*
- The forward diffusion process is defined by Hoogeboom et al. (2021);

$$q(s_j^t \mid s_j^{t-1}) = \text{Multinomial}\left((1 - \beta_t^{\text{type}}) \cdot \text{onehot}(s_j^{t-1}) + \beta_t^{\text{type}} \cdot \frac{1}{20} \cdot \mathbf{1}\right)$$

  - onehot - a function converts amino acid types to 20 dimentional vector
  - β_t^{type} is the probability of resampling

Method – X – Diffusion Process

- *Multimodel Diffusion for Amino Acid Types*
- The generative diffusion process is defined as;

$$p(s_j^{t-1} \mid R^t, C) = \text{Multinomial}\left(F(R^t, C)[j]\right)$$

  - $F(R^{\mathrm{t}}, C)[j]$ is a neural network model
    - The structure context and CDR state are taken from previous step as an input
    - It predicts the probability amino acid type for *j-th* amino acid in CDR

Method – XI – Diffusion Process

- *Diffusion for Cα Coordinates*
- The coordinates of the $C\alpha$ atoms are scaled and shifted
  - The distribution aligns more closely with a standard normal distribution
- The *forward diffusion* for normalized $C\alpha$ coordinate x_j;

$$q\left(\mathbf{x}_j^t \mid \mathbf{x}_j^{t-1}\right) = \mathcal{N}\left(\mathbf{x}_j^t \middle| \sqrt{1-\beta_{\text{pos}}^t} \cdot \mathbf{x}_j^{t-1}, \beta_{\text{pos}}^t \boldsymbol{I}\right)$$

$$q\left(\mathbf{x}_j^t \mid \mathbf{x}_j^0\right) = \mathcal{N}\left(\mathbf{x}_j^t \middle| \sqrt{\bar{\alpha}_{\text{pos}}^0} \cdot \mathbf{x}_j^0, (1-\bar{\alpha}_{\text{pos}}^0) \boldsymbol{I}\right)$$

  - β_{pos}^t controls the rate of diffusion

Method – XII – Diffusion Process

- *Diffusion for Cα Coordinates*
- *The generative diffusion* process by Ho et al. is defined as;

$$p\left(\mathbf{x}_j^{t-1}\middle|\mathcal{R}^t,\mathcal{C}\right)=\mathcal{N}\left(\mathbf{x}_j^{t-1}\middle|\boldsymbol{\mu}_p\left(\mathcal{R}^t,\mathcal{C}\right),\beta_{\text{pos}}^t\boldsymbol{I}\right),$$

$$\boldsymbol{\mu}_p\left(\mathcal{R}^t,\mathcal{C}\right)=\frac{1}{\sqrt{\alpha_{\text{pos}}^t}}\left(\mathbf{x}_j^t-\frac{\beta_{\text{pos}}^t}{\sqrt{1-\bar{\alpha}_{\text{pos}}^t}}G(\mathcal{R}^t,\mathcal{C})[j]\right).$$

  - G$(R^{\text{t}}, C)[j]$ is neural networks that predict the standard Gaussian noise

Method – XIII – Diffusion Process

- *SO(3) Denoising for Amino Acid Orientations*
- The formulation differs from diffusion models but follow similar ideas.

$$q\left(\mathbf{O}_j^t \mid \mathbf{O}_j^0\right) = \mathcal{IG}_{\mathrm{SO}(3)}\left(\mathbf{O}_j^t \middle| \mathrm{ScaleRot}\left(\sqrt{\bar{\alpha}_{\mathrm{ori}}^t}, \mathbf{O}_j^0\right), 1 - \bar{\alpha}_{\mathrm{ori}}^t\right)$$

  - $IG_{\mathrm{SO}(3)}$ denotes the isotropic Gaussian distribution on SO(3)
  - The $\mathrm{ScaleRot}$ modifies the rotation matrix
  - β_{ori}^t is the variance

Method – XIV – Diffusion Process

- *SO(3) Denoising for Amino Acid Orientations*
- *The conditional distribution used for generative process is defined as;*

$$p\left(\mathbf{O}_j^{t-1}\middle|\mathcal{R}^t,\mathcal{C}\right)=\mathcal{IG}_{\mathrm{SO}(3)}\left(\mathbf{O}_j^{t-1}\middle|H(\mathcal{R}^t,\mathcal{C})[j],\beta_{\mathrm{ori}}^t\right)$$

  - H$(R^{\mathrm{t}},C)[j]$ is neural networks that denoises the orientation

Method – XV – Neural Networks

- The use of neural network architectures for the diffusion process in modeling CDR states.
- The goal is to encode the CDR state and denoise three components:
  - amino acid types (F)
  - positions (G)
  - orientations (H)
- Multiple Layer Perceptrons (MLPs) for encoding
  - They are used to generate embeddings for individual amino acids and pairs.
    - The single one encodes amino acid type, torsional angles, and 3D coordinates
    - The pairwise one captures the distances and dihedral angles between amino acid pairs.

Method – XVI – Neural Networks

- IPA Network
  - The IPA (orientation-aware roto-translation invariant) network transforms the embeddings into hidden representations and captures the environment of each amino acid.

- Denoising MLPs
  - Three MLPs denoise:
    - Amino acid types: Outputs a 20-dimensional vector for posterior probabilities.
    - 3D positions: Predicts changes in Cα coordinates
    - Orientations: Predicts a SO(3) vector

Method – XVII – Reconstruction of CDR

- After the iterative updates, reconstructing the CDR structure at atomic level.
- Side-Chain Packing Algorithms
  - The algorithm takes the predicted orientations and positions of amino acids and pack them into a stable 3D structure.
  - The generated CDRs are not only sequence-optimized but also structually viable.

Evaluation Metrics

- Amino Acid Recovery Rate (AAR)
  - It measures the sequence identity between the reference CDR sequences and the generated CDR sequences.
- Root Mean Square Deviation (RMSD)
  - This metric assesses the structural similarity between the generated CDR structures and the original CDR structures.
- Interface Binding Energy Improvement (IMP)
  - IMP is the percentage of designed CDRs that exhibit lower (better) binding energy (ΔG) compared to the original CDRs.

Results - I

Sequence-Structure Co-design

Table 1 Evaluation of the generated antibody CDRs (sequence-structure co-design) by RAbD and the DiffAb model.

CDR	Method	AAR	RMSD	IMP
H1	RAbD	22.85%	2.261Å	43.88%
	DiffAb	65.75%	1.188Å	53.63%
H2	RAbD	25.50%	1.641Å	53.50%
	DiffAb	49.31%	1.076Å	29.84%
H3	RAbD	22.14%	2.900Å	23.25%
	DiffAb	26.78%	3.597Å	23.63%

CDR	Method	AAR	RMSD	IMP
L1	RAbD	34.27%	1.204Å	46.81%
	DiffAb	56.67%	1.388Å	45.58%
L2	RAbD	26.30%	1.767Å	56.94%
	DiffAb	59.32%	1.373Å	49.95%
L3	RAbD	20.73%	1.624Å	55.63%
	DiffAb	46.47%	1.627Å	47.32%

Results - II

Fix-Backbone Sequence Design and Structure Prediction

Table 2 Comparison of FixBB and DiffAb in terms of amino acid recovery (AAR) in the fix-backbone CDR design task.

CDR	Method	AAR	CDR	Method	AAR
H1	FixBB	37.14%	L1	FixBB	33.80%
	DiffAb	87.83%		DiffAb	86.63%
H2	FixBB	43.08%	L2	FixBB	28.54%
	DiffAb	79.70%		DiffAb	88.91%
H3	FixBB	30.74%	L3	FixBB	17.92%
	DiffAb	59.48%		DiffAb	78.69%

Results - III

Antibody Optimisation

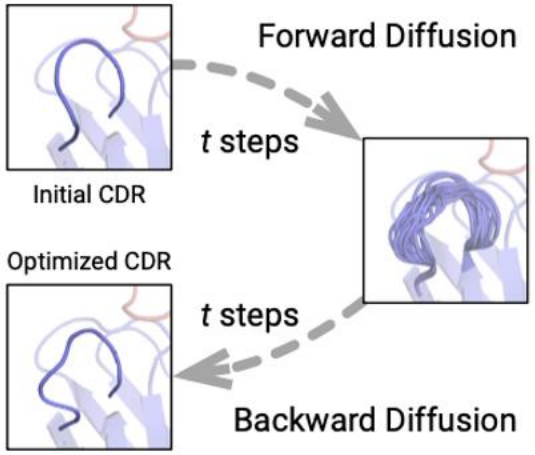

Figure 5: The antibody optimization algoritm.

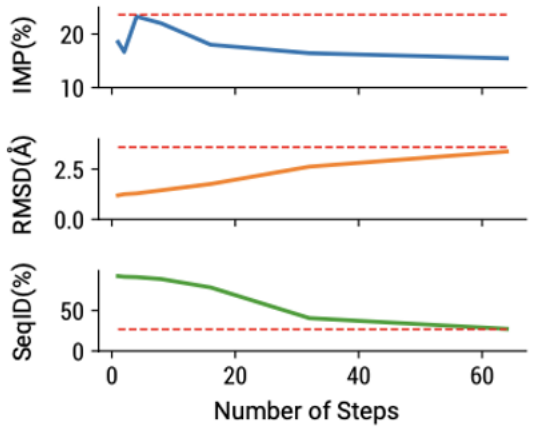

Figure 6: IMP, RMSD, and SeqID of the CDRs optimized with different numbers of steps.

Conclusion

- Uses diffusion-based generative models to enhance the specificity and quality of antibody design.
- Combines CDR sequences with their 3D structures to ensure better compatibility with target antigens.
- Takes amino acid orientations into account, critical for accurate antibody-antigen interaction modeling.
- Iteratively refines amino acid types, positions, and orientations, improving exploration of the design space.
- Optimizes current antibodies to enhance binding affinity.

References

- Codes and the data are available at: https://github.com/luost26/diffab.
- Rahmad Akbar, Habib Bashour, Puneet Rawat, Philippe A Robert, Eva Smorodina, Tudor-Stefan Cotet, Karine Flem-Karlsen, Robert Frank, Brij Bhushan Mehta, Mai Ha Vu, et al. Progress and challenges for the machine learning-based design of fit-for-purpose monoclonal antibodies. In Mabs, volume 14, page 2008790. Taylor & Francis, 2022a.
- Koichiro Saka, Taro Kakuzaki, Shoichi Metsugi, Daiki Kashiwagi, Kenji Yoshida, Manabu Wada, Hiroyuki Tsunoda, and Reiji Teramoto. Antibody design using lstm based deep generative model from phage display library for affinity maturation. Scientific reports, 11(1):1–13, 2021.
- Wengong Jin, Jeremy Wohlwend, Regina Barzilay, and Tommi S. Jaakkola. Iterative refinement graph neural network for antibody sequence-structure co-design. In International Conference on Learning Representations, 2022.
- Emiel Hoogeboom, Didrik Nielsen, Priyank Jaini, Patrick Forré, and Max Welling. Argmax flows and multinomial diffusion: Learning categorical distributions. Advances in Neural Information Processing Systems, 34, 2021
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems, 33:6840–6851, 2020.

THANK YOU