KAIST
School of Computing

Contrastive learning for antibody representation learning and antibody classification targeting pathogenic viruses

병원성 바이러스에 대응하는 항체 설계를 위한 대조적 학습

project authored by

Anar Rzayev

supervised by

Luiz Felipe Vecchietti and HyunKyu Jung (정현규)

데이터사이언스그룹
Data Science Group

Contents

1 Introduction **2**

1.1 Background 2

1.2 Traditional Methods 4

1.3 Deep Learning-based Methods 5

2 Research Objectives **6**

3 Methodology **8**

3.1 Datasets 8

3.1.1 Structural Antibody Database (SAbDab) 8

3.1.2 Observed Antibody Space (OAS) 9

3.1.3 Antigen-specific repertoire 10

3.2 Structural dataset augmentation 11

3.3 State-of-Art Approaches 11

4 Future Work **13**

References **14**

Chapter 1

Introduction

1.1 Background

The adaptive immune system of vertebrates is capable of mounting robust responses to a broad range of potential pathogens. Critical to this flexibility are antibodies, which are specialized to recognize a diverse set of molecular patterns with high affinity and specificity. The overall role of an antibody is to bind to an antigen, e.g., a virus, present it to the immune system, and stimulate an immune response. This natural role in the defense against pathogens, e.g. SARS-COV-2, Influenza, makes antibodies an increasingly popular choice for the development of new therapeutics.

An antibody consists of a heavy chain and a light chain, each composed of a variable domain (VH/VL) and a constant domain, as shown in Fig. 1.1. The variable domain is further divided into a framework region and three complementarity-determining regions (CDRs). The three CDRs on the heavy chain are denoted as CDR-H1, CDR-H2, CDR-H3, each occupying a contiguous subsequence in the framework region sequence. As the most variable part of an antibody, CDRs are the main determinants of binding and neutralization. Following current state-of-art approaches in computational biology [4-9], we formulate antibody design as a CDR generation problem, conditioned on the framework region (Fab) sequence.

Currently, monoclonal antibodies make up a rapidly growing segment of the global pharmaceutical market. The global therapeutic monoclonal antibody market was valued at approximately $150 billion in 2019 and is expected to generate revenue of $300 billion by

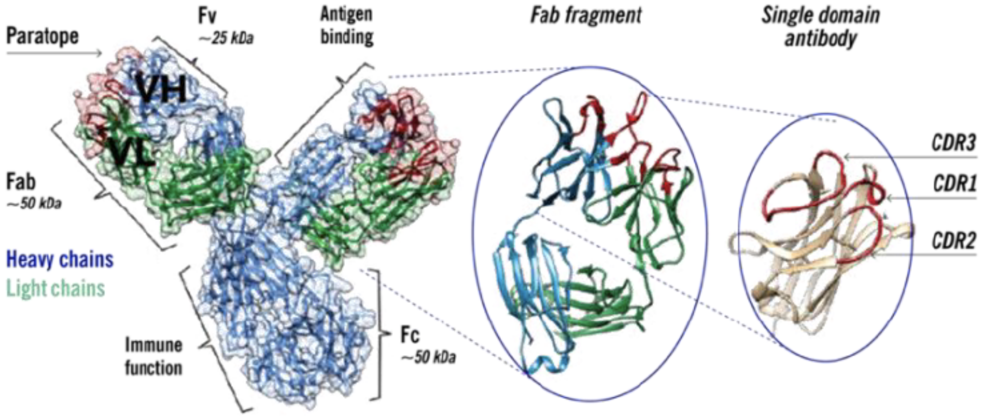


Figure 1.1: Structures of an antibody and of common antibody fragments. An antibody structure is shown on the left with the heavy (H) and light (L) chains in blue and green, respectively. CDRs containing the paratopes are coloured in red, and the heavy and light variable domains (VH and VL) are labelled. The antigen-binding fragment (Fab) region is responsible for recognising the target, while the crystallisable fragment (Fc) region for immune function and lysosome escape. The three CDR loops are highlighted in red on VH domain.

the end of 2025 [13]. However, rational design of antibody-antigen interactions is hindered by reliance on experimental methods such as crystallography, NMR, and cryo-EM, which are low throughput and requires significant investments of time and resources that may fail (Fig. 1.2).

In general, the needs for effective research or traditional diagnostics of antibodies may require in vitro selection from large, well-designed antibody libraries conducted on animal-derived experiments [5]. However, since each of different in vitro display systems, computational antibody libraries, and animals-based experiments perform differently, it makes almost impossible to predict in advance which platform will be most successful for any particular therapeutic target. Consequently, many drug discovery programs today use parallel approaches based on computational tools and machine learning to generate antibody therapeutics, increasing the chances leads will be found and reducing the time required [14]. Therefore, it is expected that research on novel computational methods are going to be crucial for the development of safer and cost-efficient therapeutic antibodies in the near future.

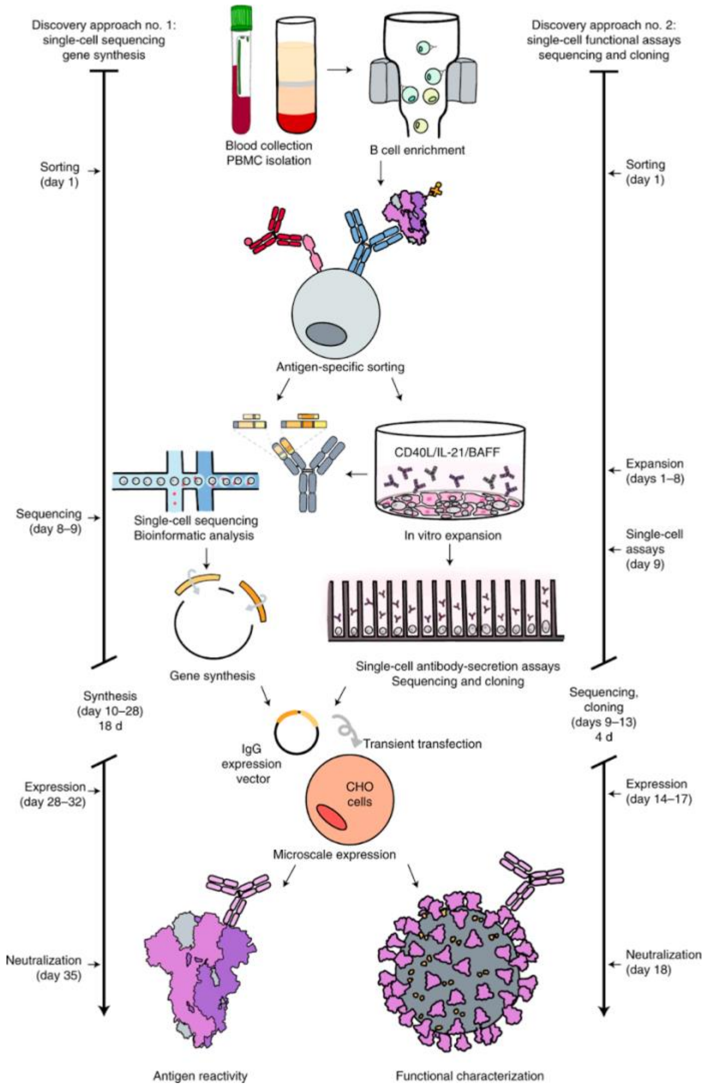


Figure 1.2: Rational design of antibody-antigen interactions

1.2 Traditional Methods

In general, methods for computational antibody design roughly fall into two categories. The first class is based on energy function optimization, which uses Markov Chain Monte Carlo simulation to iteratively modify an antibody sequence and its structure to reach a local minimum energy for the antibody structure and the interface between antibody and antigen (Fig. 1.3). Similar approaches are also used in protein design [3, 4]. However, these physics-based methods are computationally expensive, in which the designed sequence can fold into a structure different from the designed structure, and our antigen-conditioned objective can be more complicated than evaluating only physics-based binding energy models [5, 9].

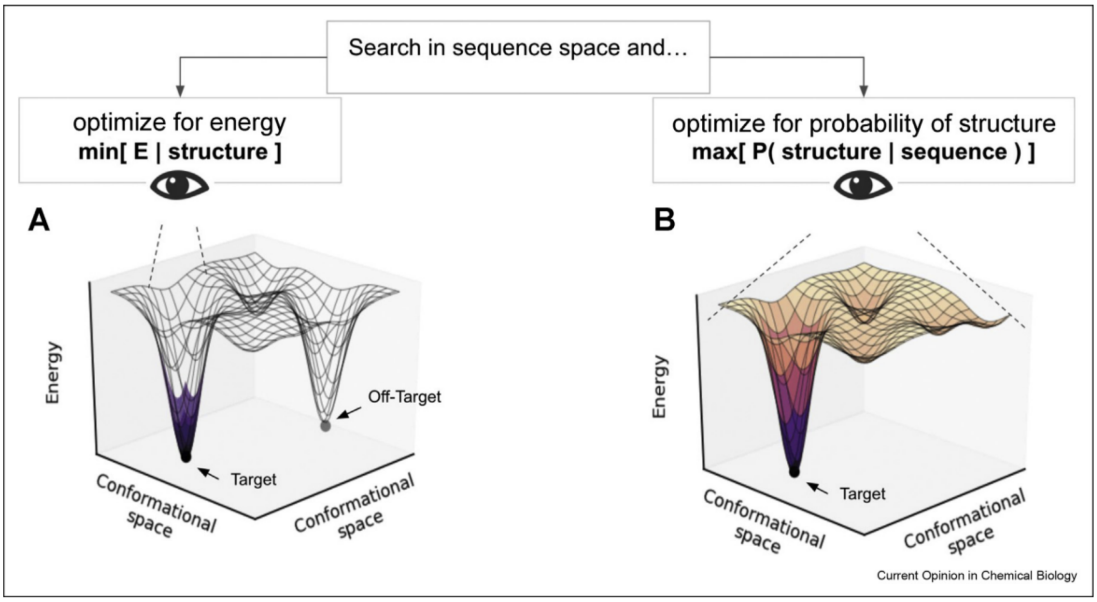


Figure 1.3: Energy Optimization of Protein Sequences

1.3 Deep Learning-based Methods

The second methodology is based on generative models. For antibodies, they are mostly sequence-based [15, 16], whereas regarding the proteins, authors from [17, 18, 19] further developed models conditioned on a backbone structure or protein folding in general. Since the best CDR structures are often unknown for new pathogens, most approaches co-design sequences and structures for specific properties of targeted viruses, mimicking auto-regressive models for graph generation.

In fact, the application of Natural Language Processing (NLP) algorithms, such as Transformers [20], and Graph Neural Networks (GNN) have been proven to be efficient for designing de-novo antibody sequences and structures [6, 7, 8, 9]. Particularly, due to the limited antibody structures available in the structural antibody database [1, 2], most of the recent research studies have focused on antibody sequence generation for paired immunoglobulin sequences using Bidirectional Encoder Representations from Transformers (BERT) models [7, 8, 12] and (self-)supervised learning algorithms [6, 9]. However, there has been a lack of studies that have investigated both structure-based and sequence-level representations to filter positive candidates from an antibody dataset which may possibly bind and neutralize specific pathogenic viruses.

Chapter 2

Research Objectives

In this work, our aim is to use contrastive learning methods (Fig. 2.1) for learning effective antibody representation (embeddings). In addition, we also intend to propose a proper data augmentation for antibody sequences and their respective structural data.

Contrastive learning has recently achieved good results to classify images in computer vision tasks [10, 11]. Using such self-supervised algorithms, the loss is designed to maximize the difference between positive examples and negative examples. In our study, specifically, positive examples could be regarded as antibodies that bind and neutralize to a specific pathogenic virus, while negative examples would be antibodies which do not bind to the specific antigen. We hypothesize that representation learned by contrastive learning using structural-based and sequence-based data can help in the classification of possible binders for specific antigens and learn good 3D/sequence representations for antigen-specificity tasks.

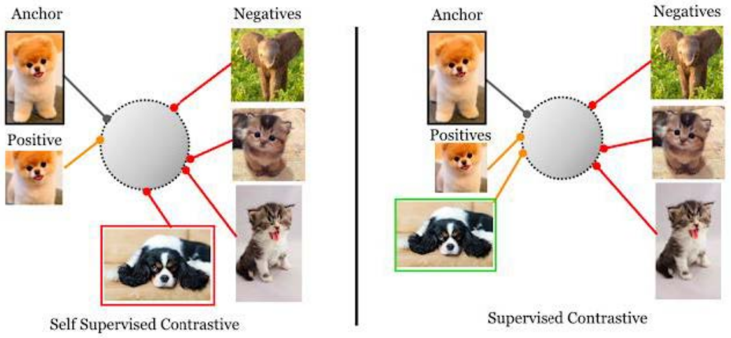


Figure 2.1: Supervised vs self-supervised contrastive losses. Supervised contrastive learning considers different samples from the same class as positive examples, in addition to augmented versions

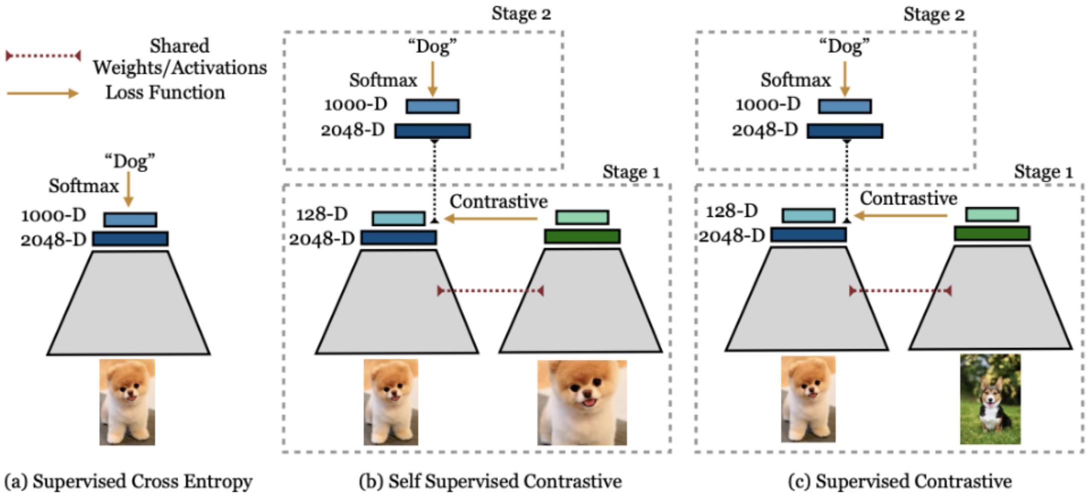


Figure 2.2: Cross entropy, self-supervised contrastive loss and supervised contrastive loss: The cross entropy loss (left) uses labels and a softmax loss to train a classifier; the self-supervised contrastive loss (middle) uses a contrastive loss and data augmentations to learn representations. The supervised contrastive loss (right) also learns representations using a contrastive loss, but uses label information to sample positives in addition to augmentations of the same image. Both contrastive methods can have an optional second stage which trains a model on top of the learned representations.

Our main objectives during this work can be summarized as follows:

- Generate a synthetic antibody structure dataset by using State-Of-The-Art (SOTA) deep learning-based antibody structure prediction networks.

- Train a classifier via contrastive learning to identify if an antibody is a potential binder to a target antigen by using sequence-based and structure-based representations (similar to Fig 2.2).

- Propose a novel data augmentation mechanism for antibody data to produce additional synthetic sequences/structures for learning better representations using contrastive learning algorithms.

- Generate novel antigen-specific antibody candidates with traditional computational methods such as Rosetta Antibody Design (RAbD) [3] and deep learning-based generative methods. Evaluate the candidates in silico using the contrastive learning-based classifier, and select the best ones for real wet-lab experiments to compute binding affinity, solubility and developability parameters.

Chapter 3

Methodology

One of the important aspects of therapeutic antibody design is to collect antigen-specific data that could be processed and filtered for efficient learning representations. As we aim to train a (self-)supervised contrastive classifier using both sequence and structure, the first step is to gather those sequence and structure data for antibodies that are confirmed to neutralize target antigens. In the next subsection, we detail how we are collecting the antibody sequences and structures to create a dataset and how we are inferring synthetic structures for antibodies that only contain sequence data using deep learning-based antibody structure prediction networks.

3.1 Datasets

3.1.1 Structural Antibody Database (SAbDab)

Structural antibody database [2] is an online resource containing all the publicly available antibody structures annotated and presented in a consistent fashion. The data are annotated with several properties including experimental information, gene details, correct heavy and light chain pairings, *antigen* details and, if available, *antibody-antigen binding affinity.* As in Fig. 3.1, the user can retrieve the full set of structures, specific entries by specifying their Protein Data Bank (PDB) code or to create subsets based on search criteria [1]. Structures can be searched based on the experimental methods used to determine the structure, species of the antibody, *type of the antigen*, presence of affinity values in the annotation and presence of amino acid residues at specific sequence positions.

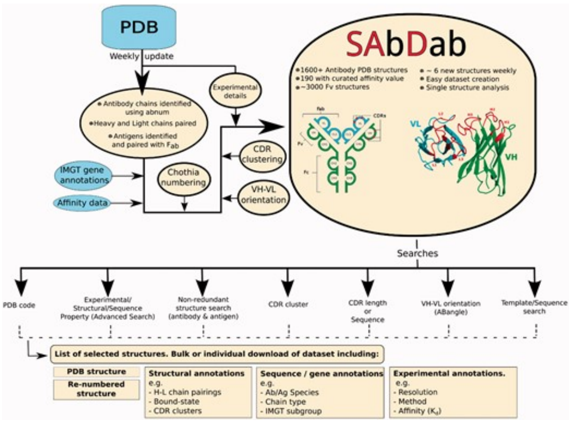

PDB
Weekly update
Experimental details
• Antibody chains identified using abnum
• Heavy and Light chains paired
• Antigens identified and paired with F_{ab}
IMGT gene annotations
Affinity data
Chothia numbering
CDR clustering
VH-VL orientation
SAbDab
• 1600+ Antibody PDB structures
• 190 with curated affinity value
• ~3000 Fv structures
• ~ 6 new structures weekly
• Easy dataset creation
• Single structure analysis
Fab
CDRs
Fv
Fc
VL
VH
Searches
PDB code
Experimental/ Structural/Sequence Property (Advanced Search)
Non-redundant structure search (antibody & antigen)
CDR cluster
CDR length or Sequence
VH-VL orientation (ABangle)
Template/Sequence search
List of selected structures. Bulk or individual download of dataset including:
PDB structure
Re-numbered structure
Structural annotations e.g. - H-L chain pairings - Bound-state - CDR clusters
Sequence / gene annotations e.g. - Ab/Ag Species - Chain type - IMGT subgroup
Experimental annotations. e.g. - Resolution - Method - Affinity (K_d)

Figure 3.1: SAbDab's workflow. New structures from the PDB are weekly analyzed to find antibody chains. These structures are then annotated with a number of properties and stored in SAbDab. Users may access and select this data using a number of different criteria. Structures and annotations can be downloaded individually or as a dataset.

3.1.2 Observed Antibody Space (OAS)

The Observed Antibody Space (OAS) database was created in 2018 to offer clean, annotated, and translated repertoire data. Driven by increasing volume of data and the appearance of paired (VH/VL) sequence data during last 4 years, OAS became accessible via a web-server [1], with standardized search parameters and sequence-based search option, to provide 1.5 billion unpaired sequences from 80 studies, including recent studies featuring SARS-CoV-2 data, and 172, 723 paired sequencing data from five studies. Providing the nucleotides for the VH/VL chains, the database also contains additional sequence annotations, such as the antibodies junction sequence and whether it is a productive sequence during wet-lab experiments, allowing for a fast initial query of 1,000 antibody sequences similar to a given sequence of interest.

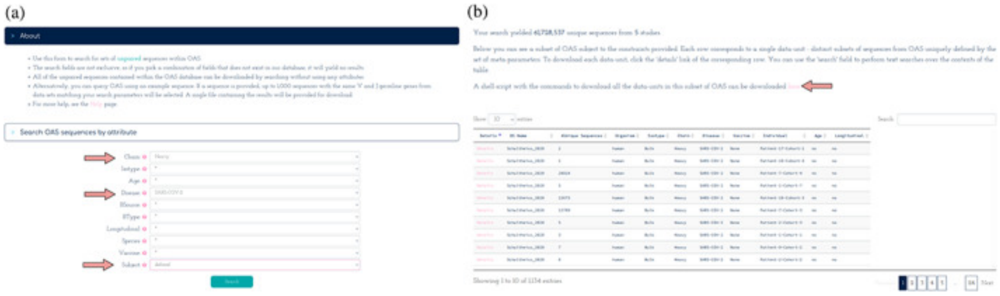
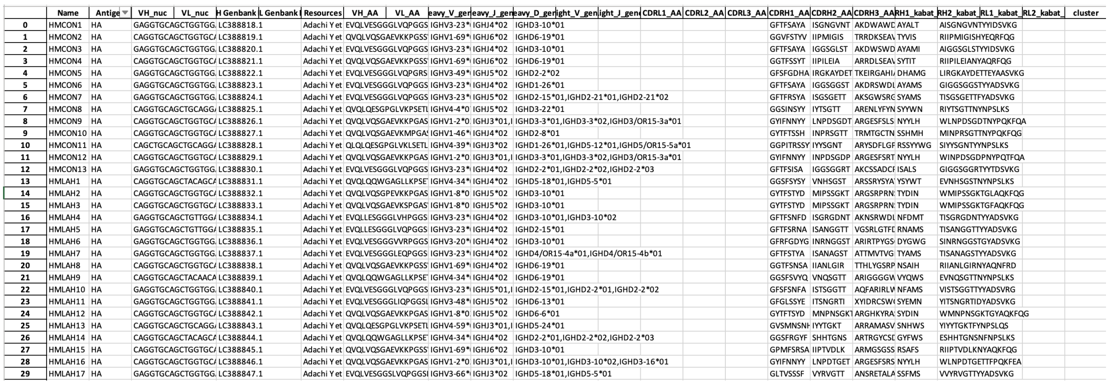


Figure 3.2: Downloading from OAS. (a) The sequence search tab for unpaired sequences, with the search options filled for heavy chain sequences from SARS-CoV-2 infected patients (shown with red arrows). (b) The search result, with each data unit matching the search and a downloadable link containing the links for the relevant data units (with a red arrow)

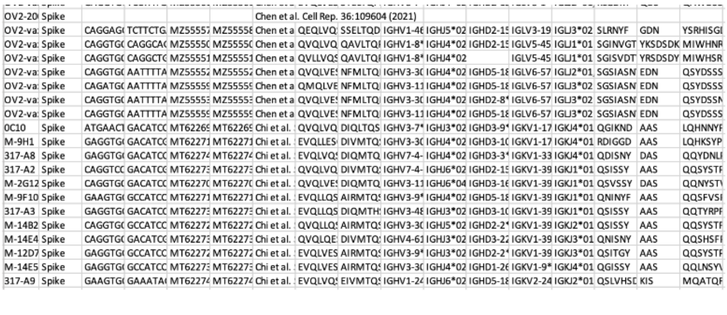
3.1.3 Antigen-specific repertoire



	Name	Antige	VH_nuc	VL_nuc	H Genbank	L Genbank	Resources	VH_AA	VL_AA	eavy_V_ger	eavy_J_ger	eavy_D_ger	ight_V_gen	ight_J_gen	CDRL1_AA	CDRL2_AA	CDRL3_AA	CDRH1_AA	CDRH2_AA	CDRH3_AA	RH1_kabat_	RH2_kabat_	RL1_kabat_	RL2_kabat_	cluster
0	HMCON1	HA	GAGGTGCAGCTGGTGG		LC388818.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ4*02	IGHD3-10*01						GFTFSAYA	ISGNGVNT	AKDWAWD	AYALT	AISGNGVNTYYIDSVKG			
1	HMCON2	HA	CAGGTGCAGCTGGTGCA		LC388819.1		Adachi Y et	QVQLVQSGAEVKKPGSS		IGHV1-69*	IGHJ6*02	IGHD6-19*01						GGVFSTYV	IIPMIGIS	TRRDKSEAV	TYVIS	RIIPMIGISHYEQRFQG			
2	HMCON3	HA	GAGGTGCAGCTGGTGG		LC388820.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ4*02	IGHD3-10*01						GFTFSAYA	IGGSGLST	AKDWSWD	AYAMI	AIGGSGLSTYYIDSVKG			
3	HMCON4	HA	CAGGTGCAGCTGGTGCA		LC388821.1		Adachi Y et	QVQLVQSGAEVKKPGSS		IGHV1-69*	IGHJ6*02	IGHD6-19*01						GGTFSSYT	IIPILEIA	ARRDLSEAV	SYTIT	RIIPILEIANYAQRFQG			
4	HMCON5	HA	GAGGTGCAGCTGGTGG		LC388822.1		Adachi Y et	EVQLVESGGGLVQPGRS		IGHV3-49*	IGHJ5*02	IGHD2-2*02						GFSFGDHA	IRGKAYDET	TKEIRGAHI	DHAMG	LIRGKAYDETTEYAASVKG			
5	HMCON6	HA	GAGGTGCAGCTGGTGG		LC388823.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ4*02	IGHD1-26*01						GFTFSAYA	IGGSGGST	AKDRSWDL	AYAMS	GIGGSGGSTYYADSVKG			
6	HMCON7	HA	GAGGTGCAGCTGGTGG		LC388824.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ5*02	IGHD2-15*01,IGHD2-21*01,IGHD2-21*02						GFTFRSYA	ISGSGETT	AKSGWSRG	SYAMS	TISGSGETTFYADSVKG			
7	HMCON8	HA	CAGGTGCAGCTGCAGGA		LC388825.1		Adachi Y et	QVQLQESGPGLVKPSETL		IGHV4-4*0	IGHJ5*02	IGHD3-22*01						GGSINSYY	IYTSGTT	ARENLYFYN	SYYWN	RIYTSGTTNYNPSLKS			
8	HMCON9	HA	CAGGTGCAGCTGGTGCA		LC388826.1		Adachi Y et	QVQLVQSGAEVKKPGAS		IGHV1-2*0	IGHJ3*01,I	IGHD3-3*01,IGHD3-3*02,IGHD3/OR15-3a*01						GYIFNNYY	LNPDSGDT	ARGESFSLS	NYYLH	WLNPDSGDTNYPQKFQA			
9	HMCON10	HA	CAGGTGCAGCTGGTGCA		LC388827.1		Adachi Y et	QVQLVQSGAEVKMPGAS		IGHV1-46*	IGHJ4*02	IGHD2-8*01						GYTFTSSH	INPRSGTT	TRMTGCTN	SSHMH	MINPRSGTTNYPQKFQG			
10	HMCON11	HA	CAGCTGCAGCTGCAGGA		LC388828.1		Adachi Y et	QLQLQESGPGLVKLSETL		IGHV4-39*	IGHJ3*02	IGHD1-26*01,IGHD5-12*01,IGHD5/OR15-5a*01						GGPITRSSY	IYYSGNT	ARYSDFLGF	RSSYYWG	SIYYSGNTYYNPSLKS			
11	HMCON12	HA	CAGGTGCAGCTGGTGCA		LC388829.1		Adachi Y et	QVQLVQSGAEVKKPGAS		IGHV1-2*0	IGHJ3*01,I	IGHD3-3*01,IGHD3-3*02,IGHD3/OR15-3a*01						GYIFNNYY	INPDSGDP	ARGESFSRT	NYYLH	WINPDSGDPNYPQTFQA			
12	HMCON13	HA	GAGGTGCAGCTGGTGG		LC388830.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ4*02	IGHD2-2*01,IGHD2-2*02,IGHD2-2*03						GFTFSISA	IGGSGGRT	AKCSSADCF	ISALS	GIGGSGGRTYYTDSVKG			
13	HMLAH1	HA	CAGGTGCAGCTACAGCA		LC388831.1		Adachi Y et	QVQLQQWGAGLLKPSE		IGHV4-34*	IGHJ4*02	IGHD5-18*01,IGHD5-5*01						GGSFSYSY	VNHSGST	ARSSRYSYA	YSYWT	EVNHSGSTNYNPSLKS			
14	HMLAH2	HA	CAGGTGCAGCTGGTGCA		LC388832.1		Adachi Y et	QVQLVQSGPEVKKPGAS		IGHV1-8*0	IGHJ5*02	IGHD3-10*01						GYTFSTYD	MIPSSGKT	ARGSRPRN	TYDIN	WMIPSSGKTGLAQKFQG			
15	HMLAH3	HA	CAGGTGCAGCTGGTGCA		LC388833.1		Adachi Y et	QVQLVQSGAEVKSPGAS		IGHV1-8*0	IGHJ5*02	IGHD3-10*01						GYTFSTYD	MIPSSGKT	ARGSRPRN	TYDIN	WMIPSSGKTGFAQKFQG			
16	HMLAH4	HA	GAGGTGCAGCTGTTGGA		LC388834.1		Adachi Y et	EVQLLESGGGLVHPGGS		IGHV3-23*	IGHJ4*02	IGHD3-10*01,IGHD3-10*02						GFTFSNFD	ISGRGDNT	AKNSRWDL	NFDMT	TISGRGDNTYYADSVKG			
17	HMLAH5	HA	GAGGTGCAGCTGTTGGA		LC388835.1		Adachi Y et	EVQLLESGGGLVQPGGS		IGHV3-23*	IGHJ4*02	IGHD2-15*01						GFTFSRNA	ISANGGTT	VGSRLGTFD	RNAMS	TISANGGTTYYADSVKG			
18	HMLAH6	HA	GAGGTGCAGCTGGTGG		LC388836.1		Adachi Y et	EVQLVESGGGVVRPGGS		IGHV3-20*	IGHJ4*02	IGHD3-10*01						GFRFGDYG	INRNGGST	ARIRTPYGS	DYGWG	SINRNGGSTGYADSVKG			
19	HMLAH7	HA	GAGGTGCAGCTGGTGG		LC388837.1		Adachi Y et	EVQLVESGGGLEQPGGS		IGHV3-23*	IGHJ4*02	IGHD4/OR15-4a*01,IGHD4/OR15-4b*01						GFTFSTYA	ISANAGST	ATTMVTVG	TYAMS	TISANAGSTYYADSVKG			
20	HMLAH8	HA	CAGGTGCAGCTGGTGCA		LC388838.1		Adachi Y et	QVQLVQSGAEVKKPGSS		IGHV1-69*	IGHJ4*02	IGHD6-19*01						GGTFSNSA	IIANLGIR	TTHLYGSRP	NSAIH	RIIANLGIRNYAQNFRD			
21	HMLAH9	HA	CAGGTGCAGCTACAACA		LC388839.1		Adachi Y et	QVQLQQWGAGLLKPSE		IGHV4-34*	IGHJ4*02	IGHD6-19*01						GGSFSVYQ	VNQSGTT	ARIGGGGW	VYQWS	EVNQSGTTNYNPSLKS			
22	HMLAH10	HA	GAGGTGCAGCTGGTGG		LC388840.1		Adachi Y et	EVQLVESGGGLVQPGGS		IGHV3-23*	IGHJ5*01,I	IGHD2-15*01,IGHD2-2*01,IGHD2-2*02						GFSFSNFA	ISTSGGTT	AQFARIRLW	NFAMS	VISTSGGTTYYADSVRG			
23	HMLAH11	HA	GAGGTGCAGCTGGTGG		LC388841.1		Adachi Y et	EVQLVESGGGLIQPGGSL		IGHV3-48*	IGHJ5*02	IGHD6-13*01						GFGLSSYE	ITSNGRTI	XYIDRCSW	SYEMN	YITSNGRTIDYADSVKG			
24	HMLAH12	HA	CAGGTGCAGCTGGTGCA		LC388842.1		Adachi Y et	QVQLVQSGAEVKKPGAS		IGHV1-8*0	IGHJ5*02	IGHD6-6*01						GYTFTSYD	MNPNSGKT	ARGHKYRA	SYDIN	WMNPNSGKTGYAQKFQG			
25	HMLAH13	HA	CAGGTGCAGCTGCAGGA		LC388843.1		Adachi Y et	QVQLQESGPGLVKPSETL		IGHV4-59*	IGHJ3*01,I	IGHD5-24*01						GVSMNSNH	IYYTGKT	ARRAMASV	SNHWS	YIYYTGKTFYNPSLQS			
26	HMLAH14	HA	CAGGTGCAGCTACAGCA		LC388844.1		Adachi Y et	QVQLQQWGAGLLKPSE		IGHV4-34*	IGHJ4*02	IGHD2-2*01,IGHD2-2*02,IGHD2-2*03						GGSFRGYF	SHHTGNS	ARTRGYCSD	GYFWS	ESHHTGNSNFNPSLKS			
27	HMLAH15	HA	CAGGTGCAGCTGGTGCA		LC388845.1		Adachi Y et	QVQLVQSGAEVKKPGSS		IGHV1-69*	IGHJ6*02	IGHD3-10*01						GPMFSRSA	IIPTVDLK	ARMGSGSS	RSAFS	RIIPTVDLKNYAQKFQG			
28	HMLAH16	HA	CAGGTGCAGCTGGTGCA		LC388846.1		Adachi Y et	QVQLVQSGAEVKKPGAS		IGHV1-2*0	IGHJ3*01,I	IGHD3-10*01,IGHD3-10*02,IGHD3-16*01						GYIFNNYY	LNPDTGET	ARGESFSRS	NYYLH	WLNPDTGETTFPQKFEA			
29	HMLAH17	HA	GAGGTGCAGCTGGTGG		LC388847.1		Adachi Y et	EVQLVESGGSLVQPGGSI		IGHV3-66*	IGHJ3*02	IGHD5-18*01,IGHD5-5*01						GLTVSSSF	VYRVGTT	ANSRETALA	SSFMS	VVYRVGTTYYADSVKG			

Figure 3.3: Examples from antigen-specificity. 2, 204 unique influenza hemagglutinin (HA) antibodies are provided with complete information for all six CDR sequences and VH/VL gene expressions

Since many sequence features of public antibody responses to different foreign viruses can be observed in Observed Antibody Space (OAS) [1] and Structural Antibody Database (SAbDab) [2], we postulate that the dataset is sufficiently large for gathering available antigen-specific antibodies for training the model. The preprocessing stage includes 172, 723 filtered paired sequences with appropriate target diseases and organisms, along with 6, 118 antibody structures available in the SAbDab. Since it is important to identify different antigens for distinguishing antibodies during the model training, we aim to retrieve 6, 273 unique SARS-CoV-2 antibodies from OAS [1], 5, 547 unique Human Immunodeficiency Virus (HIV) antibodies, and 2, 204 unique influenza hemagglutinin (HA) antibodies from GenBank [22], with complete information for all six CDR sequences and germline expressions. Among different antigens, those were mainly chosen because of large number of published sequences and expressible antibodies binding to them.



OV2-20	Spike					Chen et al. Cell Rep. 36:109604 (2021)										
OV2-va	Spike	CAGGAG	TCTTCTG	MZ55557	MZ55558	Chen et a	QEQLVQ	SSELTQD	IGHV1-46	IGHJ5*02	IGHD2-15	IGLV3-19	IGLJ3*02	SLRNYF	GDN	YSRHISGI
OV2-va	Spike	CAGGTG	CAGGCA	MZ55550	MZ55550	Chen et a	QVQLVQ	QAVLTQI	IGHV1-8*	IGHJ4*02	IGHD2-15	IGLV5-45	IGLJ1*01	SGINVGT	YKSDSDK	MIWHNR
OV2-va	Spike	CAGGTG	CAGGCT	MZ55551	MZ55551	Chen et a	QVLLVQS	QAVLTQI	IGHV1-8*	IGHJ4*02		IGLV5-45	IGLJ1*01	SGISVDT	YRSDSDY	MIWHSR
OV2-va	Spike	CAGGTG	AATTTTA	MZ55552	MZ55552	Chen et a	QVQLVE	NFMLTQ	IGHV3-30	IGHJ4*02	IGHD5-18	IGLV6-57	IGLJ2*01,	SGSIASN	EDN	QSYDSSS
OV2-va	Spike	CAGATG	AATTTTA	MZ55559	MZ55559	Chen et a	QMQLVE	NFMLTQ	IGHV3-11	IGHJ4*02	IGHD5-18	IGLV6-57	IGLJ3*02	SGSIASN	EDN	QSYDSSS
OV2-va	Spike	CAGGTG	AATTTTA	MZ55553	MZ55553	Chen et a	QVQLVE	NFMLTQ	IGHV3-30	IGHJ4*02	IGHD2-8*	IGLV6-57	IGLJ3*02	SGSIASN	EDN	QSYDSSN
OV2-va	Spike	CAGGTG	AATTTTA	MZ55559	MZ55559	Chen et a	QVQLVE	NFMLTQ	IGHV3-11	IGHJ4*02	IGHD5-18	IGLV6-57	IGLJ3*02	SGSIASN	EDN	QSYDSSS
0C10	Spike	ATGAACT	GACATCC	MT62269	MT62269	Chi et al.	QVQLVQ	DIQLTQS	IGHV3-7*	IGHJ3*02	IGHD3-9*	IGKV1-17	IGKJ4*01	QGIKND	AAS	LQHNNYF
M-9H1	Spike	GAGGTG	GACATCC	MT62271	MT62271	Chi et al.	EVQLLES	DIVMTQ	IGHV3-30	IGHJ4*02	IGHD3-10	IGKV1-17	IGKJ4*01	RDIGGD	AAS	LQHKSYP
317-A8	Spike	GAGGTG	GACATCC	MT62274	MT62274	Chi et al.	EVQLVQ	DIQMTQ	IGHV7-4-	IGHJ4*02	IGHD3-3*	IGKV1-33	IGKJ4*01	QDISNY	DAS	QQYDNLI
317-A2	Spike	CAGGTCC	GACATCC	MT62273	MT62273	Chi et al.	QVQLVQ	DIVMTQ	IGHV7-4-	IGHJ6*02	IGHD2-15	IGKV1-39	IGKJ1*01	QSISSY	AAS	QQSYSTF
M-2G12	Spike	CAGGTG	GACATCC	MT62270	MT62270	Chi et al.	QVQLVE	DIQMTQ	IGHV3-11	IGHJ6*04	IGHD3-16	IGKV1-39	IGKJ1*01	QSVSSY	DAS	QQNYST
M-9F10	Spike	GAAGTG	GCCATCC	MT62271	MT62271	Chi et al.	EVQLLQS	AIRMTQ	IGHV3-9*	IGHJ4*02	IGHD5-18	IGKV1-39	IGKJ1*01	QNINYF	AAS	QQSFVSI
317-A3	Spike	GAGGTG	GACATCC	MT62273	MT62273	Chi et al.	EVQLLQS	DIQMTH	IGHV3-48	IGHJ3*02	IGHD3-10	IGKV1-39	IGKJ1*01	QSISSY	AAS	QQTYRPF
M-14B2	Spike	CAGGTG	GCCATCC	MT62272	MT62272	Chi et al.	QVQLLQ	AIRMTQ	IGHV3-30	IGHJ5*02	IGHD2-2*	IGKV1-39	IGKJ2*01	QSISSY	AAS	QQSYSTF
M-14E4	Spike	CAGGTG	GACATCC	MT62273	MT62273	Chi et al.	QVQLQE	DIVMTQ	IGHV4-61	IGHJ3*02	IGHD3-22	IGKV1-39	IGKJ3*01	QNISNY	AAS	QQSHSFI
M-12D7	Spike	GAGGTG	GCCATCC	MT62272	MT62272	Chi et al.	EVQLVES	AIRMTQ	IGHV3-9*	IGHJ3*02	IGHD2-2*	IGKV1-39	IGKJ3*01	QSITGY	AAS	QQSYSTF
M-14E5	Spike	GAGGTG	GCCATCC	MT62273	MT62273	Chi et al.	EVQLVES	AIRMTQ	IGHV3-30	IGHJ4*02	IGHD1-26	IGKV1-9*	IGKJ4*01	QGISSY	AAS	QQLNSYV
317-A9	Spike	GAAGTG	GAAATA	MT62274	MT62274	Chi et al.	EVQLVQ	EIVMTQ	IGHV1-24	IGHJ6*02	IGHD5-18	IGKV2-24	IGKJ2*01	QSLVHSD	KIS	MQATQF

Figure 3.4: A dataset consisting of 6, 273 SARS-CoV-2 targeting antibodies with full sequence & germline expressions

3.2 Structural dataset augmentation

We seek to train the model on as many immunoglobulin structures as possible. From the Structural Antibody Database (SAbDab) [2], we obtain $6,285$ structures consisting of paired antibodies and single-chain nanobodies. Given the remarkable success of AlphaFold for modeling both protein monomers and complexes [21], we additionally explore the use of data augmentation to produce structures for training.

To produce a diverse set of structures for data augmentation, we clustered the paired and unpaired partitions of the Observed Antibody Space [1] at 40 % and 70 % sequence identity, respectively. This clustering results in $16,100$ paired sequences and $26,900$ unpaired sequences. We predict structures for both sets of sequences using the original AlphaFold model [21].

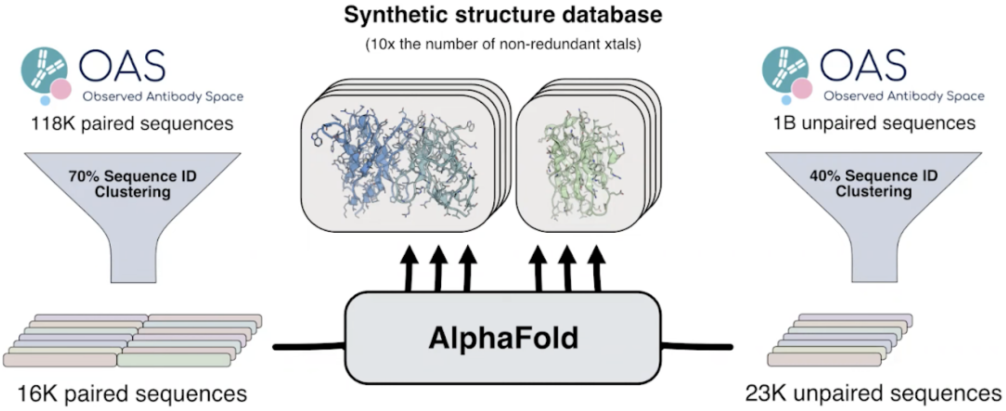


Figure 3.5: AlphaFold is used to create a synthetic structure dataset from natural antibody sequences

3.3 State-of-Art Approaches

After cleaning up redundant sequences, clustering down to more manageable population of immunoglobulins, and gathering synthetic crystal structures from antibody prediction networks [7, 8, 21], it is right time to analyze state-of-art language models incorporating both amino acid representations and structural information of antibodies.

Analysis of deep generative language models AntiBERTy [6] and IgLM [7] has demonstrated that such methods can be applied to generate synthetic libraries that may accelerate the discovery of therapeutic antibody candidates and provide new insights into antigen binding from repertoire sequences alone. However, due to the limited information from amino acid representations, it is important that structural characteristics of antibodies need to be incorporated to provide more insights about the design of immunoglobulins and possible candidates binding to specific antigens like SARS-CoV-2.

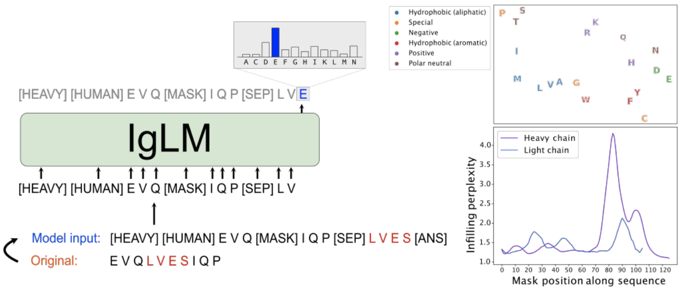

A C D E F G H I K L M N
[HEAVY] [HUMAN] E V Q [MASK] I Q P [SEP] L V E
IgLM
[HEAVY] [HUMAN] E V Q [MASK] I Q P [SEP] L V
Model input: [HEAVY] [HUMAN] E V Q [MASK] I Q P [SEP] L V E S [ANS]
Original: E V Q L V E S I Q P
Hydrophobic (aliphatic)
Special
Negative
Hydrophobic (aromatic)
Positive
Polar neutral
P S T K R Q I N H D E M L V A G Y W F C
Heavy chain
Light chain
Infilling perplexity
1.0 1.5 2.0 2.5 3.0 3.5 4.0
0 10 20 30 40 50 60 70 80 90 100 110 120
Mask position along sequence

Figure 3.6: (Left) IgLM formulates antibody design as a sequence-infilling problem based on the Infilling by Language Modeling (ILM) framework. (Top right) IgLM input layer residue embeddings cluster to reflect amino acid biochemical properties.

As mentioned in ”Research Objectives”, contrastive learning is one of the crucial self-supervised representations in classification tasks where the loss is designed to maximize difference between positive & negative samples. In this work, as we aim to solve the antigen-specificity prediction with generated in silico structural datasets, and our experimental sequence-structure examples from OAS, SAbDab, and GenBank, it is crucial to utilize embeddings from language models AntiBERTy & IgLM to reflect amino acid properties. Once the 3D learning representations from crystal structures will be obtained, this will subsequently enable to apply a more robust, accurate supervised contrastive learning model which includes both positive & negative candidates for antigen specificity.

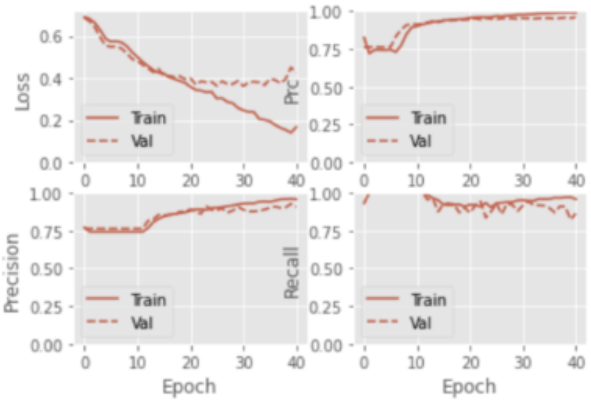


Figure 3.7: Initial results from CDR-H3 encodings [5] & Supervised Contrastive Loss [10] without data augmentation

Chapter 4

Future Work

Once we identify the epitope-paratope level interactions, binding sites, and neutralization activities available from antibody datasets, we start experimenting with the benchmark architectures for contrastive representations which will be based upon SimCSE [10] and SimCLR [11], along with appropriate graph/sequence level embeddings for antigen-specificity. Further data augmentation techniques for amino acid level representations will also be applied to improve the benchmark results of previous methods and efficiently create the antibody repertoire targeting new pathogens.

In recap, our summary for future work and ablation studies in this project can be summarized as follows:

- Obtain 3D surface-level representations of antibody structures, including in silico synthetic datasets
- Train a classifier via contrastive learning to identify if an antibody is a potential binder to a target antigen by using both sequence and structure-based representations (improving Fig. 3.7 architecture).
- Propose a novel data augmentation mechanism for antibody data to produce additional synthetic sequences/structures for learning better representations using contrastive learning algorithms.

References

[1] Tobias H Olsen, Fergus Boyles, Charlotte M Deane, 2022, Observed Antibody Space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequence

[2] James Dunbar et al, SAbDab: the structural antibody database, 2013

[3] Jared Adolf-Bryfogle, et al, 2018, RosettaAntibodyDesign (RAbD): A general framework for computational antibody design

[4] Christoffer Norn, Basile I. M. Wicky, David Juergens, Sergey Ovchinnikov, Protein sequence design by conformational landscape optimization, PNAS, 2021

[5] Yiquan Wang, Meng Yuan, Huibin Lv, Jian Peng, Ian A. Wilson, Nicholas C. Wu, A large-scale systematic survey reveals recurring molecular features of public antibody responses to SARS-CoV-2, Immunity, 2022

[6] Jeffrey A. Ruffolo, Jeffrey J. Gray, Jeremias Sulam, Deciphering antibody affinity maturation with language models and weakly supervised learning, MLSB, 2021

[7] RW Shuai, Jeffrey A. Ruffolo, Jeffrey J. Gray, Generative Language Modeling for Antibody Design, bioRxiv, 2021

[8] Jeffrey A. Ruffolo, Jeffrey J. Gray, Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies, Cell, 2022

[9] Wengong Jin, Jeremy Wohlwend, Regina Barzilay, Tommi Jaakkola, Iterative Refinement Graph Neural Network for Antibody Sequence-Structure Co-Design, arXiv, 2021

[10] Tianyu Gao, Xingcheng Yao, Danqi Chen, SimCSE: Simple Contrastive Learning of Sentence Embeddings, 2021

[11] Ting Chen, Simon Mohammad Norouzi, Geoffrey Hinton, A Simple Framework for Contrastive Learning of Visual Representations, 2020

[12] Jacob Devlin, Ming-Wei, Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv, 2018

[13] Ruei-Min Lu, Yu-Chyi Hwang, I-Ju Liu, Chi-Chiu Lee, Han-Zen Tsai, Hsin-Jung Li, Han-Chung Wu, Development of therapeutic antibodies for the treatment of diseases, PubMed Central, 2020

[14] Andrew R.M. Bradbury, Stefan Dübel, Achim Knappik, and Andreas Plückthund, Animal versus in vitro-derived antibodies: avoiding the extremes, PubMed Central, 2021

[15] Ethan C Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M Church, Unified rational protein engineering with sequence-based deep representation learning. Nature methods, 2019

[16] Jung-Eun Shin, Adam J Riesselman, Aaron W Kollasch, Conor McMahon, Elana Simon, Chris Sander, Aashish Manglik, Andrew C Kruse, and Debora S Marks, Protein design and variant prediction using autoregressive generative models, Nature communications, 2021

[17] John Ingraham, Vikas K Garg, Regina Barzilay, and Tommi Jaakkola, Generative models for graph-based protein design. Neural Information Processing Systems, 2019

[18] Alexey Strokach, David Becerra, Carles Corbi-Verge, Albert Perez-Riba, and Philip M Kim, Fast and flexible design of novel proteins using graph neural networks. BioRxiv, 2020

[19] Mostafa Karimi, Shaowen Zhu, Yue Cao, and Yang Shen. De novo protein design for novel folds using guided conditional wasserstein generative adversarial networks. Journal of Chemical Information and Modeling, 2020

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin, Attention is all you need, Advances in Neural Information Processing Systems 30 (NIPS 2017)

[21] Jumper, J., Evans, R., Pritzel, A. et al. Highly accurate protein structure prediction with AlphaFold. Nature 596, 583–589 (2021)

[22] Benson DA, Cavanaugh M, Clark K, Karsch-Mizrachi I, Lipman DJ, Ostell J, et al. GenBank. Nucleic Acids Res. 2013;41(Database issue). PubMed PMID: 23193287; PubMed Central