Deep learning algorithms for predicting association between antibody sequence, structure, and antibody properties

Constantin Schneider

Pembroke College

University of Oxford

A thesis presented for the degree of

Doctor of Philosophy

Michaelmas 2021

Abstract

Antibodies are one of the most important classes of pharmaceuticals, with over 100 antibody therapeutics approved against a wide variety of diseases and many more in active development. However, the development of antibody therapeutics remains a time- and cost-intensive process. While computational screening methods have been used to improve the efficiency of this process, they are limited by a lack of accuracy, lack of generalisability or low-throughput nature. In this thesis, I detail the development of deep learning approaches for the prediction of antibody-antigen binding, harnessing state-of-the-art approaches for the analysis of protein structures.

To this end, I start by outlining improvements to the Structural Antibody Database (SAbDab). I then describe several high-throughput deep learning tools for a structure-based antibody virtual screening pipeline using computationally generated antibody structures (DLAB). I demonstrate that convolutional neural network (CNN) models can improve the prediction of antibody-antigen complex structures through the development of the pose rescoring and docking quality assessment tool DLAB-Re. Further, by developing and evaluating the virtual screening tool DLAB-VS, I show that CNNs enable structure-based antibody-antigen binding prediction. Motivated by recent advances in protein structure analysis and equivariant graph neural network (GNN) models, I extend the DLAB framework to GNNs, developing DLAB-EG and showing that these models can offer significant advantages over existing CNN approaches.

This thesis demonstrates the applicability of state-of-the-art machine learning approaches to the antibody-antigen virtual screening task. I provide a proof-of-principle for a structure-based virtual screening pipeline for antibody model libraries as well as a promising avenue for further improvement of antibody screening capabilities using machine learning approaches.

Acknowledgements

No man is an island and despite the multitude of Covid-induced lockdowns over the course of the last years, no thesis is truly written in isolation. Therefore, I would like to use this opportunity to thank several people without whom I would have never finished this largest piece of work I have tackled in my life.

I would of course like to thank my supervisor, Professor Charlotte Deane, for giving me the chance to work on this fascinating project, being an incredibly supportive supervisor and mentor and for always encouraging me to follow interesting research directions. I would also like to thank my industrial supervisors at AstraZeneca (formerly MedImmune), Dr Bojana Popovic, Dr Bruck Taddese, Dr Andrew Buchanan, and Dr Maria Flocco, for their valuable input into my research over the last three years, as well as my funders: the EPSRC, the MRC, the SABS CDT, and AstraZeneca, for their generous financial support.

I count myself grateful for having done my DPhil in the Oxford Protein Informatics Group. The OPIGlets and particularly the members of office 2.17 have not only provided stimulating conversation, but also much needed relief in the form of all-out NERF wars and cryptic crosswording sessions.

I would like to thank my friends for brightening my time in Oxford. A special shout-out to Ari and Hannah, my flatmates during the first half of 2020, for braving the madness of the first lockdown with me and for helping to make sure that this thesis progressed despite the state of the world outside the flat door, and to Hannah and Cillian, for hiking trips and movie nights that refueled me during the final stretches.

Lastly, I would like to thank my family: My parents, who have always believed in me and who supported me throughout my life, Ich wäre ohne euch niemals so weit gekommen. My brother Valentin, who was there for me during all the highs and lows and always cheered me on. And Allie, my rock, and my biggest supporter. I would not have made it through the last three years without you. I love you.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ADC	Antibody-drug conjugate
AMA	Antibody modelling assessment
AML	Antibody model library
(R)Adam	(Rectified) Adam, an optimisation algorithm named for adaptive moment estimation
API	Application Programming Interface
BCR	B-cell receptor
CASP	Critical assessment of protein structure prediction
CAPRI	Critical Assessment of predicted interactions
CDR	Complementarity determining region
CH	Constant domain of an antibody heavy chain
CL	Constant domain of an antibody light chain
CNN	Convolutional neural network
CPU	Central processing unit
cryo-EM	Cryogenic electron microscopy
DIPS	Database of interacting protein structures
DLAB	Deep learning framework for antibodies
DLAB-EG	Deep learning framework for antibodies using equivariant graph neural networks
DLAB-Re	Deep learning framework for antibody rescoring
DLAB-VS	Deep learning framework for antibody Virtual Screening
DUD-E	The enhanced database of useful decoys
EGNN	E(n) equivariant graph neural network
Fab	Fragment antigen binding
Fc	Fragment crystallisable region
FC-layer	Fully connected layer
FDA	U.S. Food and Drug Administration
FFT	Fast Fourier transform

fnat	Fraction of native contacts
Fv	Fragment variable
GNN	Graph neural network
GPU	Graphics processing unit
HCAb	Heavy-chain only antibody
HMM	Hidden Markov model
Ig	Immunoglobulin
IMGT	International immunogenetics information system
mAb	Monoclonal antibody
MHC	Major histocompatibility complex
ML	Machine learning
MLP	Multi-layer perceptron
MSA	Multiple sequence alignment
NK-cell	Natural killer cell
NMR	Nuclear magnetic resonance
OAS	Observed antibody space
PDB	Protein data bank
PPI	Protein-protein interaction
PSA	Polar surface area
RAG	Recombination-activating genes
RBD	Receptor binding domain
ReLu	Rectified linear unit
RSS	Recombination signal sequence
(i/l)RMSD	(interface/ligand) root mean square deviation
SAbDab	The structural antibody database
SACS	The summary of antibody crystal structures database
SAS	Small angle scattering
TAP	The therapeutic antibody profiler
TCR	T-cell receptor

VH	Variable domain of an antibody heavy chain
VHH	Variable domain of a HcAB
VL	Variable domain of an antibody light chain

1 | Introduction

Contents

1.1 Introduction to antibodies

1.1.1 The role of antibodies in the immune system

Antibodies are a central component of the immune system of vertebrates. These globular, approximately Y-shaped proteins (see section 1.1.2) bind to invading pathogens, termed antigens in this context, with high affinity and specificity and induce a range of downstream effector functions, neutralising pathogens and activating other components of the immune system through the mechanisms detailed below and in Fig. 1.1 and Table 1.1.

Antibodies can neutralise soluble antigens in the bloodstream through binding to the antigen, inhibiting the interaction between the antigen and its target in the body and/or marking it for phagocytosis. Antibodies can also bind to antigens expressed on cell surfaces. This coating of cell surfaces with antibodies can lead to phagocytosis by macrophages (in which case it is referred to as opsonisation), activation of the complement pathway and subsequent introduction of pores and lysis of the cells or recruitment of natural killer (NK) cells to induce cell death. Antibodies can also sensitise mast cells, inducing ejection of pathogens, for example through coughing (Murphy and Weaver 2017).

These different effector functions are mediated through a set of antibody isotypes, which differ in their heavy chain constant region (see Fig. 1.2). There are five immunoglobulin (Ig) isotypes (IgM, IgD[1], IgG, IgA, IgE) which differ both in their effector function as well as in their

[1] While the other four isotypes are at this point well characterised, IgD is less well understood and its function remains the object of active research (Wan et al. 2021; Gutzeit et al. 2018).

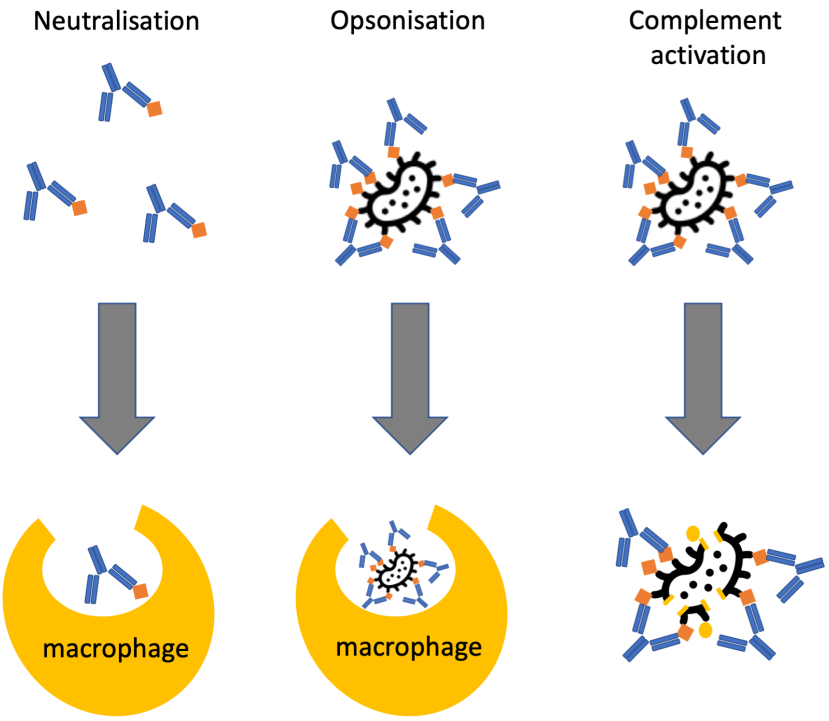

Figure 1.1: Schematic overview of antibody effector functions. Antibodies (blue) can neutralise soluble antigens (orange) by binding and marking for phagocytosis. Binding membrane-bound antigens can induce phagocytosis through opsonisation, activation of the complement system or sensitisation of the antigen displaying cell to NK-cell induced cell death.

multimerisation behaviour. IgG (which has four subtypes differing in their effector function, see Table 1.1), IgE and IgD are monomers, while IgA can form dimers and IgM can form pentamers or hexamers (Murphy and Weaver 2017).

Antibodies are able to bind specifically and with high affinity to antigens due to their large sequence diversity: the theoretical size of the antibody sequence space is 10^{18} (Rees 2020). In the bloodstream, antibodies are presented on circulating B-cells and are therefore also referred to as B-cell receptors (BCRs). Each B-cell generates and presents only one antibody, meaning that the number of distinct antibodies in circulation is limited by the number of B-cells in

Function	IgM	IgD	IgG1	IgG2	IgG3	IgG4	IgA	IgE
Neutralisation	+	-	++	++	++	++	++	-
Opsonisation	+	-	++	+	++	+	+	-
NK cell recruitment	-	-	++	-	++	-	-	-
Mast cell sensitisation	-	-	+	-	+	-	-	+++
Complement system activation	+++	-	++	+	+++	-	+	-

Table 1.1: Isotype dependent Ig effector function. No effector function is indicated "-", effector function is indicated "+", "++" and "+++" depending on strength. The table is adapted from Murphy and Weaver (2017).

circulation, which is estimated to be 10^9, nine orders of magnitude smaller than the theoretical antibody sequence space and six orders of magnitude smaller than the estimated size of the human antibody repertoire (Briney et al. 2019). Therefore, once an initial, low affinity binding event between antibody and antigen occurs, the antibody sequence space is sampled for higher affinity binders through affinity maturation (see Section 1.1.3).

1.1.2 Antibody structure

Natural human antibodies assume a roughly Y-like three-dimensional structure. They are homodimers of a heterodimer formed by a heavy chain and a light chain (named for their respective molecular weight)[2], linked by bisulfide bridges (Narciso et al. 2011). Both heavy and light chain can be subdivided into two domains: the constant domain (CH/L) which is highly conserved in terms of both sequence and structure, and the variable domain (VH/L), which contains the antigen-binding sequence variable regions. While the light chain contains one constant do-

[2]There are antibody formats which deviate from this format, these are covered in section 1.1.5

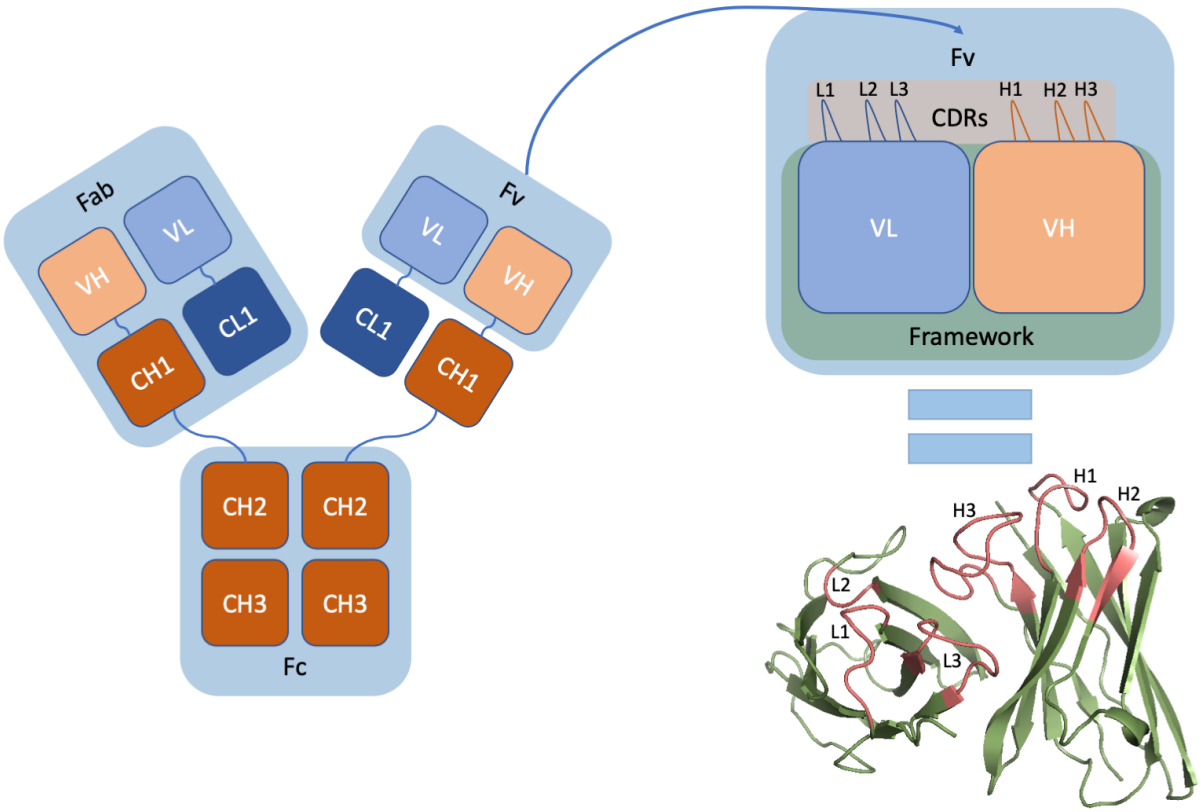

Figure 1.2: Schematic overview of the antibody structure. (**Left**) Overall structure of the antibody, with heavy and light chain constant and variable regions as well as Fv, Fc and Fab highlighted. The antibody displayed here is of IgG isotype. (**Right**) Schematic of the Fv region (top) and corresponding antibody structure (bottom). On the bottom structure, the three CDRs are coloured salmon, the four framework regions green.

main (CL1), the human heavy chain contains either three or four constant domains[3] (CH1-3/4, depending on the heavy chain isotype) and encodes the antibody effector functions described in section 1.1.1. Further, the dimerised tail of the constant domain (CH2-3/4) is referred to as the fragment crystallisable region (Fc) while the paired heavy and light chain regions (VH, VL, CH1, CL1) are referred to as the antigen binding fragment (Fab) and the paired variable regions (VH, VL) are referred to as the variable fragment (Fv), where the majority of antibody sequence diversity is located (Murphy and Weaver 2017).

The variable domain can be further subdivided into alternating framework regions and complementarity determining regions (CDRs). Like the constant domain, the four framework regions (FR 1-4) are highly conserved in both sequence and structure. The main influence the variable domain framework regions have on the variability of antibody structure and antibody-antigen binding is through their influence on the VH-VL packing orientation (Foote and Winter 1992). The relative orientation of paired heavy and light chain variable domains is not conserved across antibodies and can strongly affect the structure of the antigen-binding region on the antibody. The VH-VL orientation can be described through different sets of angles and distance, in this thesis I use the ABangle tool for this purpose, which describes the orientation with a set of five angles and one distance (Dunbar et al. 2013).

The CDRs are hypervariable regions, the variability of which is the main factor modulating antigen recognition specificity (Liu 2014). The mechanism for the generation of CDR variability is described in section 1.1.3. The CDRs are loops, able to adopt a variety of structures.

[3]There are antibodies in other animals that can deviate from this, shark IgNAR antibodies have no light chain and five constant domains (Feige et al. 2014; Fernández-Quintero et al. 2021), and antibodies consisting of only heavy chains can also be found in camelids, see section 1.1.5.

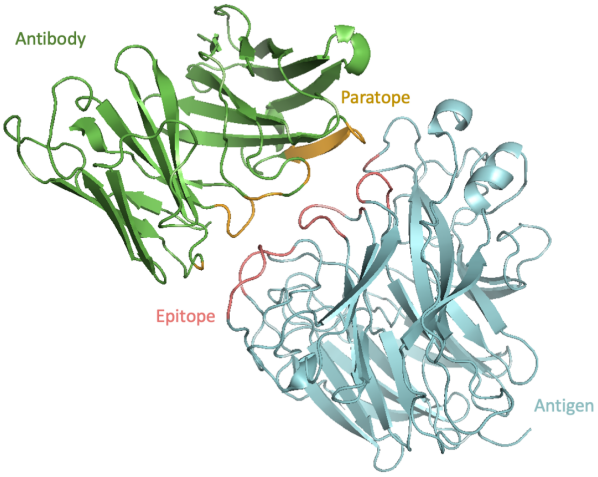

Figure 1.3: Crystal structure (PDB: 1a14) of an antibody-antigen binding pose. The antibody is coloured green, with the paratope highlighted in yellow. The antigen is coloured cyan with the epitope highlighted in salmon.

There are three CDRs in each of the light and heavy chain variable domain, referred to as L1, L2 and L3 on the light chain and H1, H2 and H3 on the heavy chain[4]. CDRs H/L1 and H/L2 have the least sequence variability, while CDR L3 is more variable and CDR H3 is the most variable. CDR H3 variability is due to the two gene junctions found in its sequence (see section 1.1.3). Owing to its high variability in structure and length, CDR H3 is often the strongest contributor to antibody-antigen binding specificity and affinity (Narciso et al. 2011). The structures of CDRs H/L1, H/L2 and L3 can be clustered into a discrete set of so-called canonical forms, which can simplify the prediction of antibody structures from sequence (see section 1.2.2) (e.g. North et al. (2011) and Wong et al. (2019)). These clusters are not exhaustive and are therefore redefined as more antibody structures are experimentally determined (Wong et al. 2019). CDR H3, due to its high variability in sequence composition and length, does not

[4]There is another solvent exposed loop on antibody chains which shows higher sequence variability than the framework regions (the DE-loop), which has been suggested as a fourth CDR (Kelow et al. 2020). However, this loop is not considered a CDR by most computational tools and studies and so is treated as framework in this thesis.

assume canonical forms.

When binding to a cognate antigen, the residues on the antibody in contact with the antigen are referred to as the paratope, while the residues on the antigen contacting the antibody are referred to as the epitope (see Fig. 1.3). The majority of paratope residues are found in the CDRs, though framework residues can sometimes contribute to the antibody-antigen binding interface (Narciso et al. 2011).

1.1.3 Antibody sequence diversity, generation, and maturation

As mentioned above, the binding of antibodies via their paratope to a wide array of epitopes is mediated through the large sequence diversity of antibodies in their variable region. The sequence of the variable region is sampled during antibody generation through several mechanisms.

Antibody heavy and light chain nucleic acid sequences are composed of several gene segments. The heavy chain gene locus, located in humans on chromosome 14, contains several constant region genes (C $\alpha/\delta/\epsilon/\gamma/\mu$), which determine the Ig isotype (IgA/D/E/G/M), and an array of gene segments from which the variable domain is assembled. These are referred to as variable (V), diversity (D) and joining (J) (Roth 2014). There are currently 57 human heavy chain V genes, 23 human D genes and 6 human heavy chain J genes catalogued in the IMGT/GENE-DB database (Giudicelli 2004). During differentiation of B-cells from progenitor cells, the expressed heavy chain variable domain nucleic acid sequence is formed through the recombination of these gene segments, first joining a V gene with a D gene to form VD, then joining

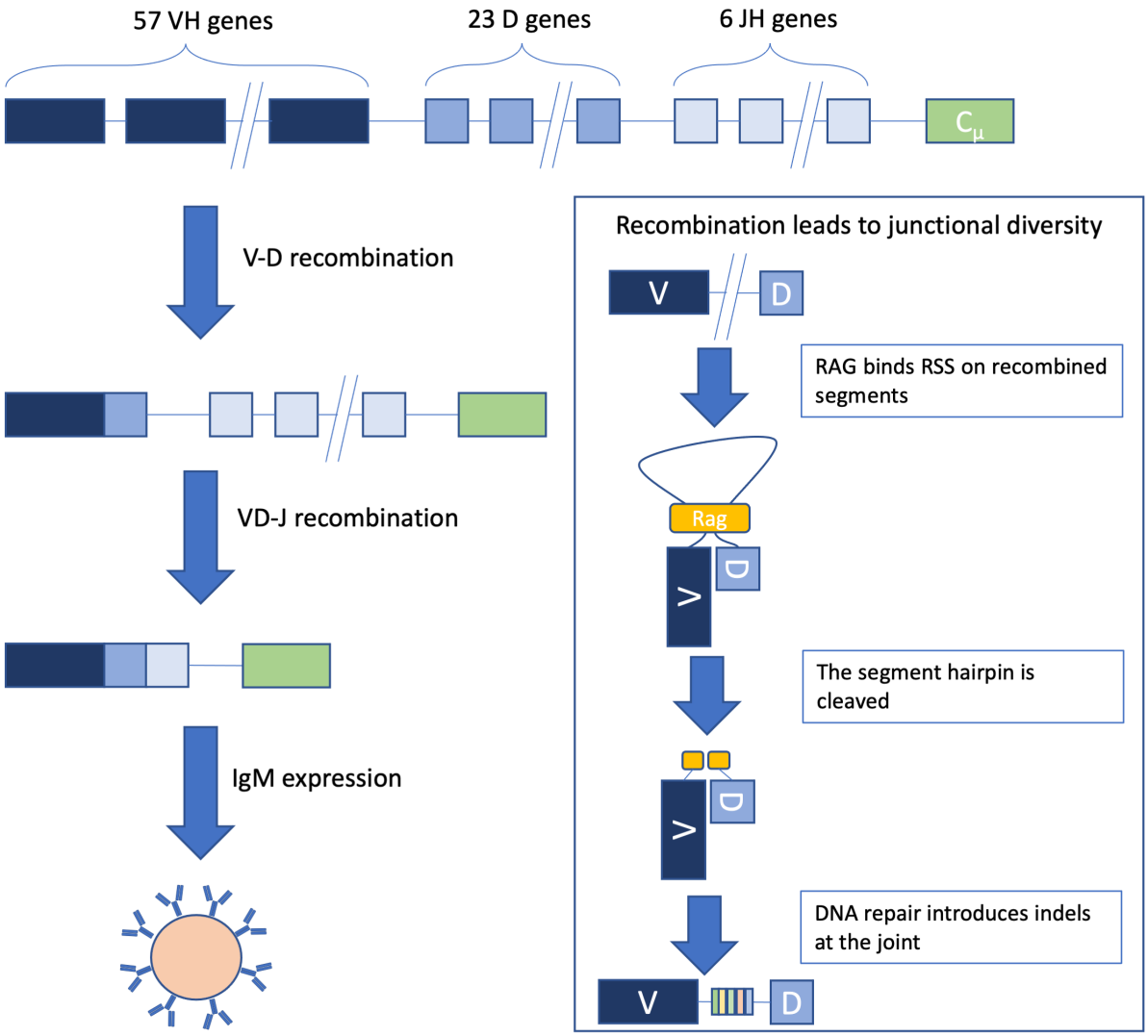

Figure 1.4: Simplified overview of antibody V(D)J recombination. The recombination of heavy chain VDJ genes is shown, the recombination process for light chain genes is equivalent, but only using one recombination event (V-J). One V and one D gene are first recombined to form the VD-pair, which is then recombined with a J gene to form the VDJ segment. The box shows the generation of junctional sequence diversity through the creation of insertions or deletions (indels) at the V-D junction. The same process occurs at the VD-J junction and at the V-J junction for light chains.

VD with a J gene to form the full variable domain heavy chain sequence VDJ (see Fig. 1.4). Light chain gene recombination follows the same process, but light chains do not have the D gene, so only one recombination event (V-J) takes place. Further, there are two light chain gene loci, kappa ($\varkappa$) and lambda (λ), on chromosome 2 and 22 respectively, of which each mature B-cell expresses one[5], which is referred to as isotypic exclusion (Murphy and Weaver 2017).

This combinatorial diversity enables the generation of a large space of distinct antibody sequences. Not counting alleles of the variable domain genes, there are 7866 possible heavy chain variable domains, 205 $\varkappa$ and 165 λ light chain variable domains (Giudicelli 2004), i.e. approximately $2.9 * 10^6$ unique Fv regions (if alleles are taken into account, this number rises to $1.6 * 10^8$)[6]. Of those, each B-cell presents one after differentiation is complete.

A second source of sequence diversity during the V(D)J recombination is the generation of junctional diversity (Roth 2014). To join two variable region genes, the RAG enzyme binds to the recombination signal sequence (RSS) flanking the genes to be recombined and creating a hairpin. The hairpin DNA is then cleaved, creating a nick at both genes. During the process of joining the two genes and repairing the nick, random insertions or deletions (indels) can occur at the junction, introducing regions of high sequence diversity. CDR 3 occupies the 3' end of V and the 5' end of J, meaning that it is the only CDR with junctional sequence diversity, explaining the higher diversity in both length and amino acid composition of CDR 3. Since only the heavy chain variable domain contains the D gene, CDR L3 gains diversity from one

[5] Dual $\varkappa$+λ expression in the same B-cell can occur but is comparatively rare (Giachino et al. 1995).

[6] A recent study demonstrated that V(D)J recombination does not follow the same probabilistic model across individuals, but rather differs on an individual basis, so the space effectively sampled differs across individuals (Slabodkin et al. 2021).

junction, while CDR H3 gains diversity from two junctions, explaining the higher sequence and length diversity of CDR H3 compared to CDR L3.

Finally, the antibody sequence space is sampled for antigen affinity during somatic hypermutation (Storb et al. 1999). Upon binding of circulating B-cells presenting IgM antibodies to an antigen (likely with low affinity) and activation by T-helper cells, B-cells migrate to the lymph nodes and form germinal centers. There, the B-cells proliferate, and activation-induced cytidine deaminase (AID) induces random mutations in the antibody sequence (by deaminating cytosine to uracil). These mutations are focused on the gene regions corresponding to the CDRs, with those regions as a result accumulating mutations at a rate several orders of magnitude higher than the baseline somatic mutation rate. If the resulting antibodies loose affinity for the antigen or start to exhibit cross-reactivity, the corresponding B-cells undergo apoptosis. B-cells with higher affinity antibodies proliferate further and undergo additional rounds of somatic hypermutation.

After affinity maturation, the Ig isotype is modulated via cytokine-induced class-switching and matured B-cells differentiate into plasmablasts, which release antibodies, or memory B-cells which stimulate antibody production upon re-exposure to the same antigen, enabling improved long-term protection against the antigen. This memory mechanism is used by vaccines to prime the immune system for exposure to antigens (Murphy and Weaver 2017).

Sequence 1	X	X	I1	I2	I3	I4	I5	X	X
Asymmetric numbering	1	2	2A	2B	2C	2D	2E	3	4
Symmetric numbering	1	2	2A	2B	2C	3B	3A	3	4
Sequence 2	X	X	I1				I2	X	X
Asymmetric numbering	1	2	2A				2B	3	4
Symmetric numbering	1	2	2A				3A	3	4

Table 1.2: Comparison of symmetric and asymmetric numbering schemes. A hypothetical symmetric and asymmetric numbering scheme is used to number a sequence of four conserved amino acids (X) with an insertion of either five or two amino acids (I) after the second conserved position. An asymmetric numbering scheme (like the Chothia scheme) assigns different identifiers to the last inserted residues, though those are likely in structurally equivalent positions, whereas a symmetric numbering scheme (like the IMGT scheme) assigns the same identifier to those residues, more accurately capturing the structural significance of insertions in antibody sequence.

1.1.4 Antibody sequence numbering

The generation of antibodies outlined above results in a high level of sequence conservation across antibodies and consistent sequence positions of variability. This conservation has been exploited to develop numbering schemes for antibody variable region sequences, which enable comparisons between antibody sequences and particularly their CDRs (Dondelinger et al. 2018).

Numbering schemes for antibody sequences are derived either from sequence alignments, structural clustering, or a combination thereof. The Kabat scheme (Kabat et al. 1979), the first standardised antibody numbering scheme, used sequence alignments to generate separate num-

bering schemes for the antibody heavy chain as well as the κ and λ light chains. The Chothia scheme (Chothia and Lesk 1987) expands upon the Kabat scheme by redefining the positions of amino acid insertions based on not just sequence, but also structure-based alignments. Abhinandan and Martin (2008) further expanded upon the Chothia scheme by including the large available volume of both sequences and structures at the time of their publication, which included sequence lengths not considered in the Kabat or Chothia numbering schemes. The Kabat scheme and its successors have two main flaws. Firstly, insertions are asymmetric, complicating the comparison of antibodies with different CDR lengths (see Table 1.2). Secondly, these schemes differ for heavy and light chain variable regions, as well as for other immunologically relevant proteins, like T-cell receptors, complicating comparative analysis. The IMGT numbering scheme (Lefranc et al. 2003) addresses these pitfalls, deriving the numbering from germline sequence alignments to be compatible across the immunoglobulin superfamily and using symmetric insertion numbering.[7] In this thesis, I therefore use IMGT numbering where possible (some processing steps used in chapter 3 required the use of Chothia numbering).

Standardised antibody numbering schemes further enable the identification of CDRs based on the numbering. The studies detailing numbering schemes often also contain suggestions for the residues considered to be part of the CDRs according to that numbering scheme (Chothia and Lesk 1987; Abhinandan and Martin 2008; Lefranc et al. 2003), which can be updated as additional data becomes available (North et al. 2011). As the boundaries for CDR definitions fall outside of insertion points and the antibody numbering schemes are bijective outside

[7] There is a variety of other numbering schemes (Dondelinger et al. 2018), but these follow the same general principles as the three described here, are less widely used, and are omitted for the sake of brevity.

of insertion points (i.e. each position in one numbering has exactly one corresponding position in another numbering scheme), CDR definitions are interchangeable between numbering schemes. In this thesis, I use both IMGT and Chothia CDR definitions.

There are several numbering tools for antibody sequences available (e.g. Li et al. (2019), Abhinandan and Martin (2008), and Dunbar and Deane (2015)). Throughout this thesis, I use the ANARCI numbering tool (Dunbar and Deane 2015), which uses Hidden Markov Models (HMMs) to assign numberings.

1.1.5 Antibodies as therapeutics

1.1.5.1 Market size and disease targets

The central role of antibodies in the immune system as well as their high sequence diversity and corresponding diversity of antigen specificity detailed in the previous sections make antibodies promising candidates for therapeutics development. Global sales in monoclonal antibody therapeutics are growing rapidly and reached an estimated $98 billion in 2018 (Grilo and Mantalaris 2019) and $122 billion in 2019 (Mullard 2021) and are projected to grow to approximately $300 billion by 2025 (see Fig. 1.5), with 100 monoclonal antibody therapeutics approved by the FDA as of May 2021 (Mullard 2021). Antibody therapeutics are used against a variety of disease targets, including rheumatoid arthritis (adalimumab), cancer (e.g. pembrolizumab), Crohn's disease (ustekinumab), osteoporosis (denosumab) and SARS-CoV-2 (imdevimab). The antibody therapeutics development process is covered in more detail in section 1.4.

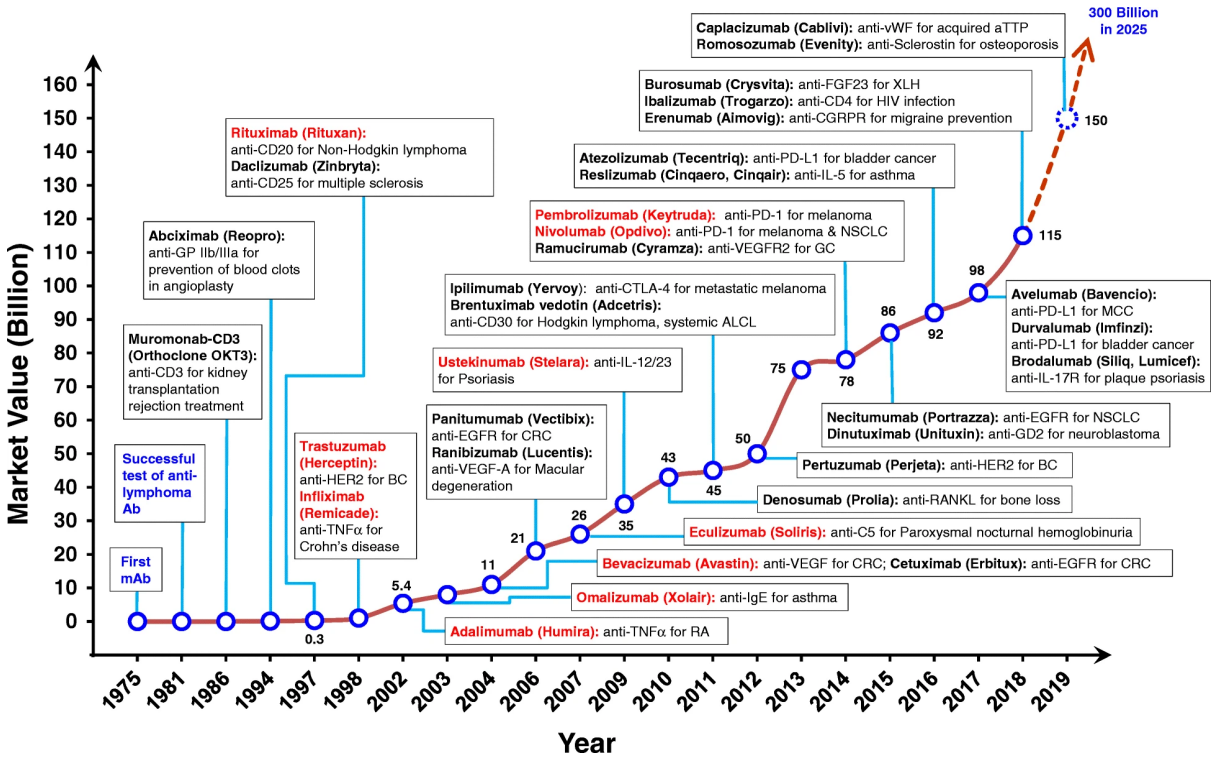

Figure 1.5: Timeline from 1975 showing the successful development of therapeutic antibodies and their applications. Many biotech companies that promised antibodies as anticancer "magic bullets" were launched from 1981 to 1986. The height of the line and numerical annotations represent the estimated market value of mAb therapeutics in each indicated year (shown as billions of US dollars). Antibodies coloured in red represent the top 10 best-selling antibody drugs in 2018.

Figure and caption text reproduced from Lu et al. (2020) under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/).

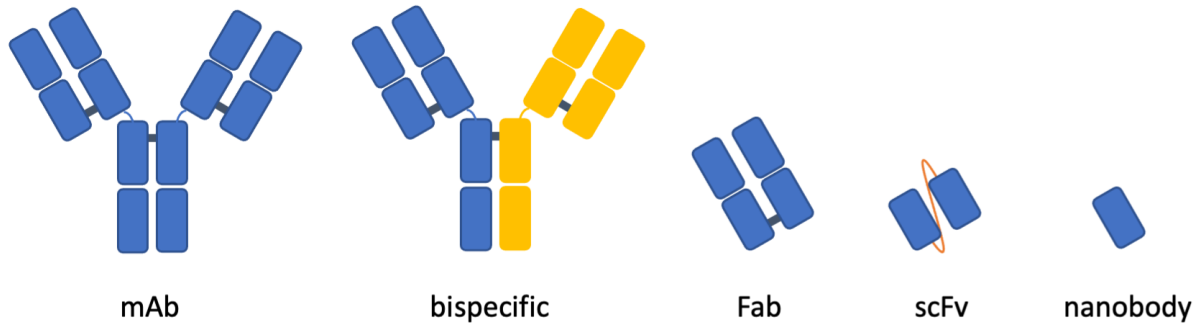

Figure 1.6: Common therapeutic antibody formats. The formats are explained in detail in section 1.1.5.2.

1.1.5.2 Antibody therapeutic formats

The majority of antibody therapeutics, including all those listed above as examples, are monoclonal antibodies (mAbs), comprising the entire antibody molecule, with both paratopes binding the same antigen. Most monoclonal antibodies are of IgG isotype, though there has been recent interest in exploring the use of different isotypes for mAbs therapeutic development (Samsudin et al. 2020; Sterlin and Gorochov 2021). Monoclonal in this context refers to the fact that the antibody therapeutics are generated from clones of one B-cell (in contrast to polyclonal antibodies).

Despite their success as therapeutics, the high molecular weight of mAbs can cause challenges during production and can lead to reduced tissue penetration. Due to the presence of the Fc region in mAbs, they can also be overly immunogenic in patients (Bates and Power 2019). Therefore, several antibody fragment formats have been developed for use as therapeutics (see Fig. 1.6).

The Fab format consists of only the Fab region (the VL, CL, VH and CH1 domains, see sec-

tion 1.1.2), linked at the base through a bisulfide bridge. An approved example of that format is ranibizumab, targeting Vascular Endothelial Growth Factor A. The single-chain Fv (scFv) format completely foregoes constant domains, consisting of only a VH and a VL domain, connected by a linker. An approved example is brolucizumab, also targeting Vascular Endothelial Growth Factor A. Nanobodies are a fragment format that has received increased attention in recent years (Yang and Shah 2020; Jovčevska and Muyldermans 2020; Krah et al. 2016), including as potential therapeutics against SARS-CoV-2 (Huo et al. 2020). These VH-only fragments are based on antibodies first discovered in camelids (Hamers-Casterman et al. 1993), which consist of a dimer of heavy chains (HCAb). The variable domain of HCAbs (VHH) has since been repurposed as a standalone therapeutics candidate, also referred to as nanobody. Nanobodies benefit from their extremely low molecular weight (approximately 15 kDa) compared to mAbs (approximately 150 kDa), enabling easier manufacturing, tissue penetration and crossing of the blood-brain barrier (Yang and Shah 2020). Further, nanobody CDR3 loops are longer than corresponding antibody CDR H3 loops (Desmyter et al. 2001), enabling higher affinity binding to target epitopes and the exploration of epitope topologies inaccessible to conventional antibody paratopes (Holliger and Hudson 2005). As a result, there is large interest in developing nanobody therapeutics, with one nanobody therapeutic (caplacizumab) currently approved and six in clinical trials (Raybould et al. 2020). In chapter 2, I describe the creation of a dedicated resource for the collection and analysis of nanobody structures.

Naturally occurring antibodies bind to the same epitope with both antigen binding sites. Bispecific antibody formats combine different binding sites to generate therapeutics able to bind two different epitopes simultaneously. This can be used to either enable highly specific binding to

target cells by binding to two epitopes on the same target, which can also enable antibody therapeutics disrupting multi-component signalling pathways, or to recruit immune cells to targets of interest, commonly used in bispecifics for cancer therapy (Ma et al. 2021). There are over 100 bispecific formats in development or clinical trial (Ma et al. 2021) and three bispecifics have been approved so far (amivantamab, blinatumomab, emicizumab) (Raybould et al. 2020).

A special case of antibody therapeutic formats are antibody-drug conjugates (ADCs) (Beck et al. 2017). In this format, an antibody (or antibody-fragment) is linked to a payload, a cytotoxic small molecule drug. Their primary application is in oncology, where there are currently nine approved ADCs (Drago et al. 2021).

There are many variations on the antibody therapeutic formats described here in development (pre-clinical or clinical) or already approved as therapeutics (Bates and Power 2019; Spiess et al. 2015; Labrijn et al. 2019), but the specifics of those formats go beyond the scope of this thesis.

1.2 Generation of antibody structures

The analysis of antibody-antigen binding and the development of structure-based tools for antibody virtual screening such as those described in this thesis requires three dimensional structures of antibodies and particularly of the Fv region, which drives antibody-antigen binding. There are two approaches to the generation of those structures: experimental determination of the structure of *in-vitro* expressed antibodies or computational prediction of antibody structure from the amino acid sequence.

1.2.1 Experimental determination of antibody structures

1.2.1.1 X-ray crystallography

Since the first experimental determination of a protein structure (myoglobin in 1957), X-ray crystallography has been the major method of general protein structure determination as well as antibody structure determination: to date, 83% of antibody structures have been determined using X-ray crystallography (Dunbar et al. 2014). While X-ray crystallography reliably enables the generation of high-quality structures, expressing and crystallising proteins is a laborious process involving iterative improvement of crystallisation conditions. Modelling of protein structures from X-ray readouts is also time-intensive. Taken together, this leads to a comparatively slow increase in the number of determined crystal structures: while other experimental areas, most notably gene sequencing (Cook et al. 2016), have seen exponential increases in the volume of data deposited, the rate of antibody crystal structure determination has grown much slower, from 138 antibody structures determined using X-ray crystallography in 2010 to 551 in 2020, an approximately 4-fold increase.

1.2.1.2 Cryo-EM

Over the last 20 years, cryogenic electron microscopy (cryo-EM) has seen a rapid increase in popularity as a protein structure determination method (Benjin and Ling 2020; Callaway 2015). For this approach[8], a thin layer of concentrated solution of the protein of interest is

[8]For the purpose of this introduction, I ignore alternative imaging approaches using cryo-EM, such as cryo-EM tomography for whole cell imaging (Turk and Baumeister 2020) and focus on the most common method, single particle cryo-EM.

imaged under an electron microscope after freezing in liquid nitrogen. The resulting image consists of thousands of 2D projections of the protein from different angles, which can be used to recreate the 3D structure in Fourier space (Cheng et al. 2015).

While cryo-EM does not require the crystallisation of the protein of interest, the preparation of the protein in solution for imaging can be challenging as the imaged protein layer needs to be of single-molecule thickness while also containing many distinct orientations of the protein in order to enable the accurate recreation of the 3D protein structure (Lyumkis 2019), which can lead to sample preparation times similar to those required for X-ray crystallography. Cryo-EM however has the advantage of capturing the protein in solution (if not at *in-vivo* temperature), as opposed to X-ray crystallography, in which the crystallised protein structure can differ from the structure in solution (Garbuzynskiy et al. 2005). While early cryo-EM structures had low resolution (Binshtein and Ohi 2015), the current structure determination pipelines now routinely produce structures of similar quality to high resolution crystal structures (e.g. Nakane et al. 2020). This is reflected in the fact that while in the period between the deposition of the first cryo-EM antibody structure in 1999 and the start of the research detailed in this thesis in 2018, only 4% of antibody structures were determined using cryo-EM, 34% of antibody structures deposited since then were cryo-EM structures as of November 2021 (Dunbar et al. 2014).

1.2.1.3 NMR and other methods

There are several other methods for protein structure determination, including nuclear magnetic resonance (NMR) (Cavalli et al. 2007) and small angle scattering (SAS) (Mertens and Svergun 2010), which, while of interest to general protein structure determination, are beyond the scope

of this introduction, as antibody structures determined using these methods only constitute 0.5% of all antibody structures as of November 2021 (Dunbar et al. 2014).

1.2.1.4 Deposition and cataloguing of antibody structures

Since its inception in 1971, experimentally determined protein structures have been deposited by the community in the Protein Data Bank (PDB) (Berman et al. 2000), which now contains over 180,000 macromolecular structures, including all publicly available experimentally determined antibody structures. However, as the PDB has not been created with the specific requirements of antibody structural analysis in mind, several databases of experimentally determined antibody structures have been created to enable the creation of bespoke antibody data sets and the analysis of structural features of antibodies[9]:

The Summary of Antibody Crystal Structures (SACS) database (Allcorn and Martin 2002) is a monthly updated data set tabulating PDB deposited antibody structures, identified via sequence identity to other antibody structures and alignment to antibody germlines. Alongside information derived from the PDB header, each entry contains antibody chain isotypes and antigen information (where applicable) alongside CDR sequences.

IMGT/3Dstructure-DB is the 3D structure database of the international ImMunoGenetics information system (IMGT) (Ehrenmann et al. 2010). It contains information on all antibody as well as T-cell receptor (TCR) and major histocompatibility complex (MHC) protein structures. Each entry in IMGT/3Dstructure-DB is annotated with information derived from the IMGT/GENE-

[9]In this introduction, I only cover antibody structure databases which are currently accessible online and are actively updated. ABDB (Ferdous and Martin 2018) for example, while still being accessible, has not been updated since 2019.

DB database (Giudicelli et al. 2004) according to the IMGT ontology (Giudicelli and Lefranc 1999), alongside paratope and epitope annotation and allows the direct download of the PDB file renumbered with the IMGT numbering scheme (see section 1.1.4).

The Structural Antibody Database (SAbDab) (Dunbar et al. 2014) is a weekly updated database of antibody structures which, alongside advanced search features, enables the creation and bulk download of bespoke antibody structure data sets. The latter feature makes SAbDab particularly suited for structural analysis of antibody-antigen interactions as well as training set creation for machine learning (ML) models and it has been used for that purpose both by me and by other researchers (e.g. Schneider et al. (2021b), Robinson et al. (2021), Hou et al. (2021), and Myung et al. (2021)). Antibodies are identified from each weekly PDB update and parsed to be annotated with PDB header information, amino acid sequence including the complementarity determining region (CDR), the heavy and light chain subgroup, antigen type, and, where available, antigen binding affinity value and can be downloaded with either Chothia or IMGT numbering applied to the antibody chains. SAbDab also has a python API which enables more fine-grained analysis of the database contents, and which was used to compile the data sets of experimentally determined structures throughout this thesis. In chapter 2, I describe several improvements made to SAbDab which improve the ability of users to create and maintain custom data sets.

More narrowly defined antibody structure databases have been created to curate data sets for specific research communities: CoV-AbDab (Raybould et al. 2021a) catalogues sequences and structures of all known coronavirus-binding antibodies. Thera-SAbDab (Raybould et

al. 2020) collects all World Health Organisation-recognised therapeutic antibodies alongside (where available) experimentally determined structures. In chapter 2 I further describe the creation of SAbDab-nano, a sub-database of SAbDab tracking all nanobodies and VH-only antibodies deposited in the PDB.

1.2.2 Computational prediction of antibody structures

1.2.2.1 The protein structure prediction task

As detailed in section 1.2.1, while experimental methods are able to generate high-quality protein structures, they do so at a low pace compared to the growth of available amino acid sequence information. This is particularly true for antibodies: due to the combinatorial antibody generation process outlined above (see section 1.1.3), the antibody sequence space is multiple orders of magnitudes larger than the space of experimentally resolved antibody structures (Rees 2020). The Observed Antibody Space (OAS) database is an annotated database of immune-repertoire sequencing studies (Kovaltsuk et al. 2018), containing over half a billion sequences, while only $\sim$ 5600 antibody structures have been deposited in SAbDab as of November 2021. Effective exploration of structural antibody space, either for the purpose of identifying novel therapeutic candidates, to gain insights into the underlying chemistry and biology, or to generate structure-based computational tools, therefore requires the computational generation of accurate structural models from antibody sequences.

The prediction of protein structure from amino acid sequence is a well characterised problem, with a plethora of tools released over the roughly 50 years it has been investigated (Kuhlman

and Bradley 2019; Pearce and Zhang 2021). To enable unbiased comparison of structure prediction tools, a biannual blind assessment of protein structure prediction tools, the Critical Assessment of Structure Prediction (CASP) experiment, has been performed since 1994 (Moult et al. 1995). Historically, the protein structure prediction problem was divided into two primary approaches: template-based prediction, where a structure with high sequence homology is available and can be used as a template for modelling, and template-free prediction, where no template is available and the prediction tool instead has to predict the structure *ab-initio*. Of those, template-based predictions tended to give considerably more accurate results (Pearce and Zhang 2021). Over the last two CASP competitions, template-free modelling tools incorporating deep learning approaches into their pipeline have enabled a sharp increase in the quality of predicted structures (Kryshtafovych et al. 2019; Kryshtafovych et al. 2021), with CASP14 reporting the ground-breaking performance of the AlphaFold2 model (Jumper et al. 2021), which achieved prediction accuracy within the uncertainty of the experimentally determined structures for the majority of CASP target structures and has lead the CASP organisers to state that ”the problem of computing atomic accuracy protein structures from amino acid sequence is solved" (Kryshtafovych et al. 2021). Importantly, the performance of AlphaFold2 did not require good homology templates for the target proteins to be available, while also yielding improved predictions over template-based modelling tools for targets where good homology templates existed (Jumper et al. 2021).

AlphaFold2 has since been used to model multimeric proteins (Evans et al. 2021) and heterodimer complexes (Yin et al. 2021; Bryant et al. 2021) (see section 1.6.2.3), to sample alternative protein conformational states (Del Alamo et al. 2021), and to create a large data set of

predicted protein structures, including 98.5% of the human proteome (Tunyasuvunakool et al. 2021).

Beyond a highly optimised model architecture, AlphaFold2's success can be attributed to several concepts which have since been widely adopted by the field (and in related areas of protein structural analysis, see section 1.5). While deep learning on multiple sequence alignments (MSAs) for structure prediction has been used in state-of-the-art protein structure prediction models previously (Pearce and Zhang 2021), AlphaFold2 uses an end-to-end learning model. Previous state-of-the-art approaches used MSAs to generate geometric restraints for physics-based downstream structure optimisation, while AlphaFold2 directly generates an output structure[10]. Further, AlphaFold2 was the first method using a network architecture equivariant to rotation and translation on this task (see section 1.5.3.2 for more detail on equivariance).

Other machine learning tools for general protein structure prediction using approaches similar to AlphaFold2 (though less successfully) have been published between the announcement of the performance of AlphaFold2 at CASP 14 and the publication of the AlphaFold2 method (Baek et al. 2021).

1.2.2.2 AlphaFold2 in the context of antibody modelling

While AlphaFold2 performs well on general protein structure prediction, strongly outperforming other approaches, the same is not true for antibody modelling (Abanades et al. 2021). Here, the native AlphaFold2 architecture as published by Jumper et al. (2021) models antibodies with

[10]While AlphaFold2 was not the first structure prediction approach using end-to-end learning (AlQuraishi 2019), it was the first such method achieving state-of-the-art performance and predicting atom coordinates directly, as opposed to torsion angles (Pearce and Zhang 2021)

acceptable accuracy but performs worse than both non-ML tools as well as domain specific ML tools for the task. While the AlphaFold2 model trained to predict multimeric complexes (Evans et al. 2021) achieves better performance on the antibody modelling task, it still performs worse than domain specific tools (personal communication with Brennan Abanades). Further, there is anecdotal evidence to suggest that AlphaFold2 achieves lower accuracy on framework residues than current homology modelling approaches (personal communication with Carlos Outeiral), which is of particular interest given the high structural conservation of that region. Lastly, AlphaFold2 is compute intensive, requiring approximately 3h to predict one antibody structure on a 20-core CPU (Abanades et al. 2021), as opposed to 30s for ABodyBuilder on a single CPU core (Leem et al. 2016), making it poorly suited for high-throughput applications such as the virtual screening tools detailed in chapters 4 and 5.

To overcome these shortcomings of AlphaFold2 for antibody modelling, domain-specific tools using similar approaches to AlphaFold2 have been recently released (Abanades et al. 2021; Ruffolo et al. 2020; Ruffolo et al. 2021; Zenkova et al. 2021). Abanades et al. (2021) for example developed ABlooper, an equivariant graph neural network (see section 1.5.3.2) which can predict CDR loops quickly (0.05s per sequence) and with state-of-the-art accuracy and can be used in conjunction with homology modelling tools.

No comprehensive study into the reasons for the reduced performance of AlphaFold2 on antibodies compared to general protein structures has been so far undertaken, but this difference is likely because AlphaFold2 performs less well on proteins where only shallow MSAs are available (Jumper et al. 2021). Deep MSAs can be constructed for antibodies due to the high conser-

vation in most of the sequence, but in the CDRs, particularly in CDR H3, these MSAs contain little co-evolutionary information. For the purpose of prediction using the AlphaFold2 architecture, the antibody CDRs therefore do not benefit from the MSA depth, potentially causing the poor prediction in those areas. Further, there is little scope for improvement of modelling of antibody framework regions, which can already be homology-modelled with sub-angstrom RMSD (Leem et al. 2016).

Throughout the research detailed in this thesis, including after the publication of AlphaFold2, I use modelling tools specifically developed for the antibody domain.

1.2.2.3 Antibody structural modelling

Due to the high sequence and structure conservation in antibodies outside of the CDRs, most antibody modelling tools follow the same overall workflow to model the antibody variable domain (Norman et al. 2020): initially, the framework regions 1-4 (see Fig. 1.2) for both heavy and light chain are modelled from homology templates. Then, the packing orientation of heavy and light chain is predicted and a model of the Fv framework is constructed. Following on from the framework prediction, the CDRs are predicted, usually by identifying the canonical structures of the CDRs L1-3 and H1-2 and predicting the CDR H3 by template-based modelling, *ab-initio* modelling or a combination thereof. Lastly, side chains are predicted, and any additional optimisation steps are performed.

ABodyBuilder (Leem et al. 2016) is a state-of-the-art antibody structure modelling tool which I have used for antibody structure prediction throughout the research detailed in this thesis.

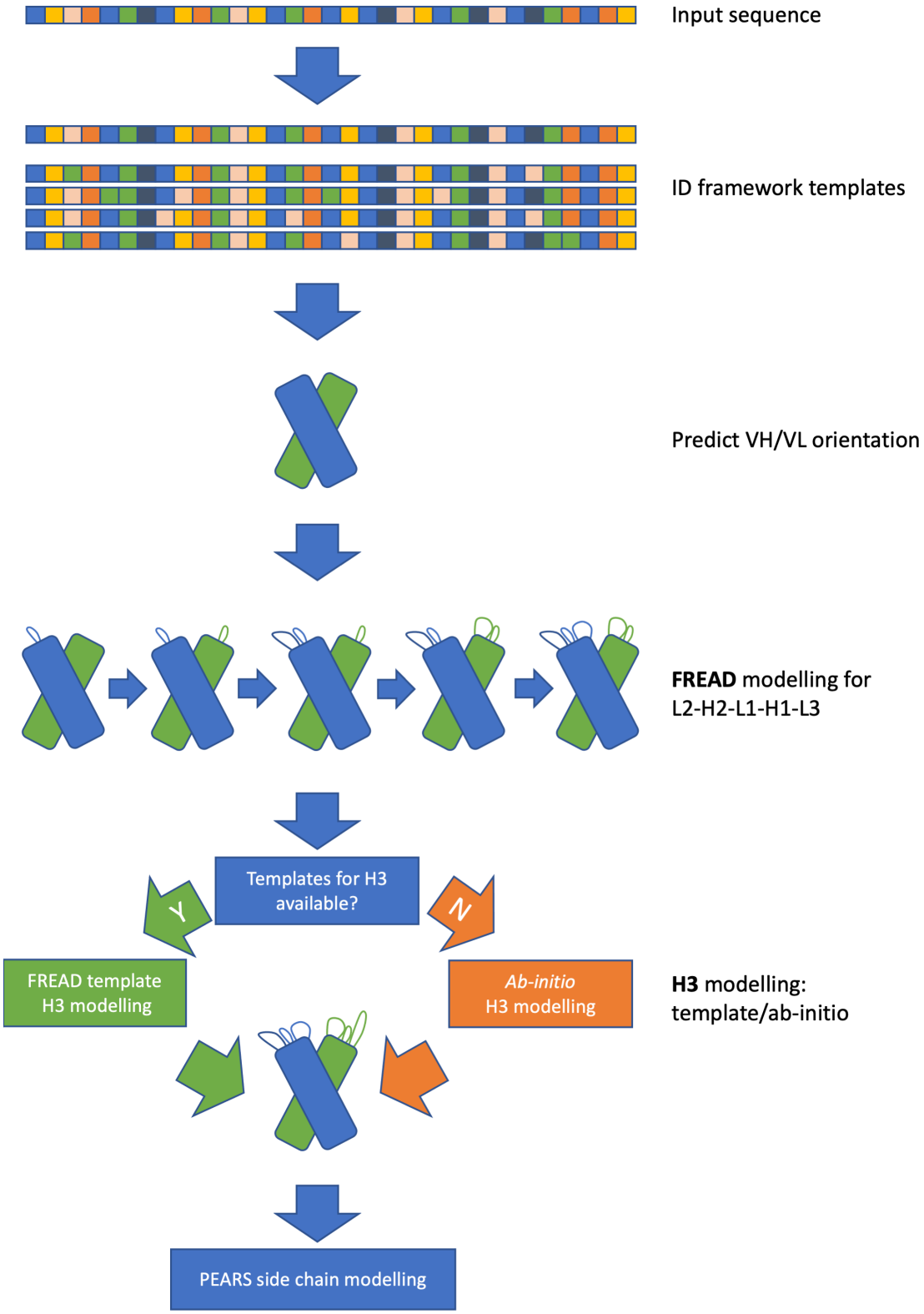

Figure 1.7: ABodyBuilder pipeline. Caption on next page.

Figure 1.7: continued from previous page

The overall workflow shown here is shared by most state-of-the-art modelling tools. Template sequences are identified and used to generate a preliminary heavy and light chain framework. Using the templates, the VH/VL orientation is predicted and applied to the framework templates. CDR loops are modelled using FREAD where templates of sufficient quality and identity are available or *ab-initio* modelled where such templates are not available (this is shown for H3, which is most likely to be *ab-initio* modelled). Lastly, side chains are modelled using PEARS. The light chain is schematically coloured in blue, the heavy chain in green.

It implements the steps outlined above as shown in Fig. 1.7: initially, the query sequence is numbered using ANARCI. Framework templates are picked individually for heavy and light chain based on framework sequence identity. If the same template is chosen for both chains, the VH/VL orientation of that template is adopted for the prediction. Otherwise, a consensus template is identified for VH/VL orientation, and the heavy and light chain templates are oriented according to the ABangle parameters (Dunbar et al. 2013) of the consensus template. CDRs are then modelled in the order L2-H2-L1-H1-L3-H3 using FREAD (Choi and Deane 2010) for template-based prediction (applying a series of cut-offs of decreasing stringency to identify suitable templates) and MODELLER (Šali and Blundell 1993) or Sphinx (Marks et al. 2017) for *ab-initio* prediction if FREAD fails to model the CDR loop. Then, side chains are copied from the templates or predicted using PEARS (Leem et al. 2018).

As mentioned above, most state-of-the-art antibody modelling tools follow the same workflow, with differences primarily found in their CDR modelling approach (Weitzner et al. 2017; Klausen et al. 2015; Lepore et al. 2017; Schritt et al. 2019). RosettaAntibody (Weitzner et al.

2017) for example has a similar overall workflow as ABodyBuilder but incorporates energy-based refinement for all CDR modelling steps and always models the CDR H3 loop *ab-initio* and refines the final structure through VH/VL re-packing.

Similar to the biannual CASP competition detailed above, there have been community-wide assessments of antibody modelling. The second antibody modelling assessment (AMA-II) (Almagro et al. 2014) compared many of the current state-of-the-art antibody modelling tools and has since served as a benchmark set for new modelling tools[11]. These comparisons have shown that most current state-of-the-art tools achieve similar accuracy, with all tools performing poorly on the CDR H3 compared to the other CDRs (Almagro et al. 2014; Norman et al. 2020). Throughout the research detailed in this thesis, I use ABodyBuilder due to its state-of-the-art accuracy combined with its fast prediction speed (approximately 30s per antibody sequence) (Leem et al. 2016; Norman et al. 2020).

As alluded to in section 1.2.2.2, several machine learning approaches specific to the antibody structure prediction task have been published over the last two years, which are surpassing the traditional modelling approaches while maintaining prediction speed and are likely to become the standard in the field (Abanades et al. 2021; Zenkova et al. 2021; Ruffolo et al. 2020; Ruffolo et al. 2021).

[11] ABodyBuilder for example was published after AMA-II, but a version without access to structures deposited after AMA-II was created for benchmarking (Leem et al. 2016).

1.3 Computational generation of antibody-antigen complex structures

1.3.1 Introduction to docking

In order to study the interaction between antibodies and cognate antigens on a structural level as well as to classify their capacity to form binding interactions, structures of the binding interface are needed.

However, due to the limits of experimental structure generation detailed in section 1.2.1 only approximately 3200 structures of antibodies in complex with protein antigens have been experimentally determined to date. Given the size of both the antibody sequence space and the number of possible target antigens, there are trillions of possible antibody-antigen pairings, so computationally predicted complex structures are needed to assess the binding potential of a given antibody-antigen pairing.

The computational generation of protein-protein or protein-ligand complex poses (where ligand refers to a small molecule ligand) is referred to as docking. Due to the focus of this thesis being on antibody protein-antigen binding, I will focus here on protein-protein docking and refer for example to Pagadala et al. (2017) for a comprehensive review of protein-ligand docking methods.

Docking programs generate a number of interaction poses between the interaction partners. These interaction poses are then assessed with a scoring function to identify native-like poses

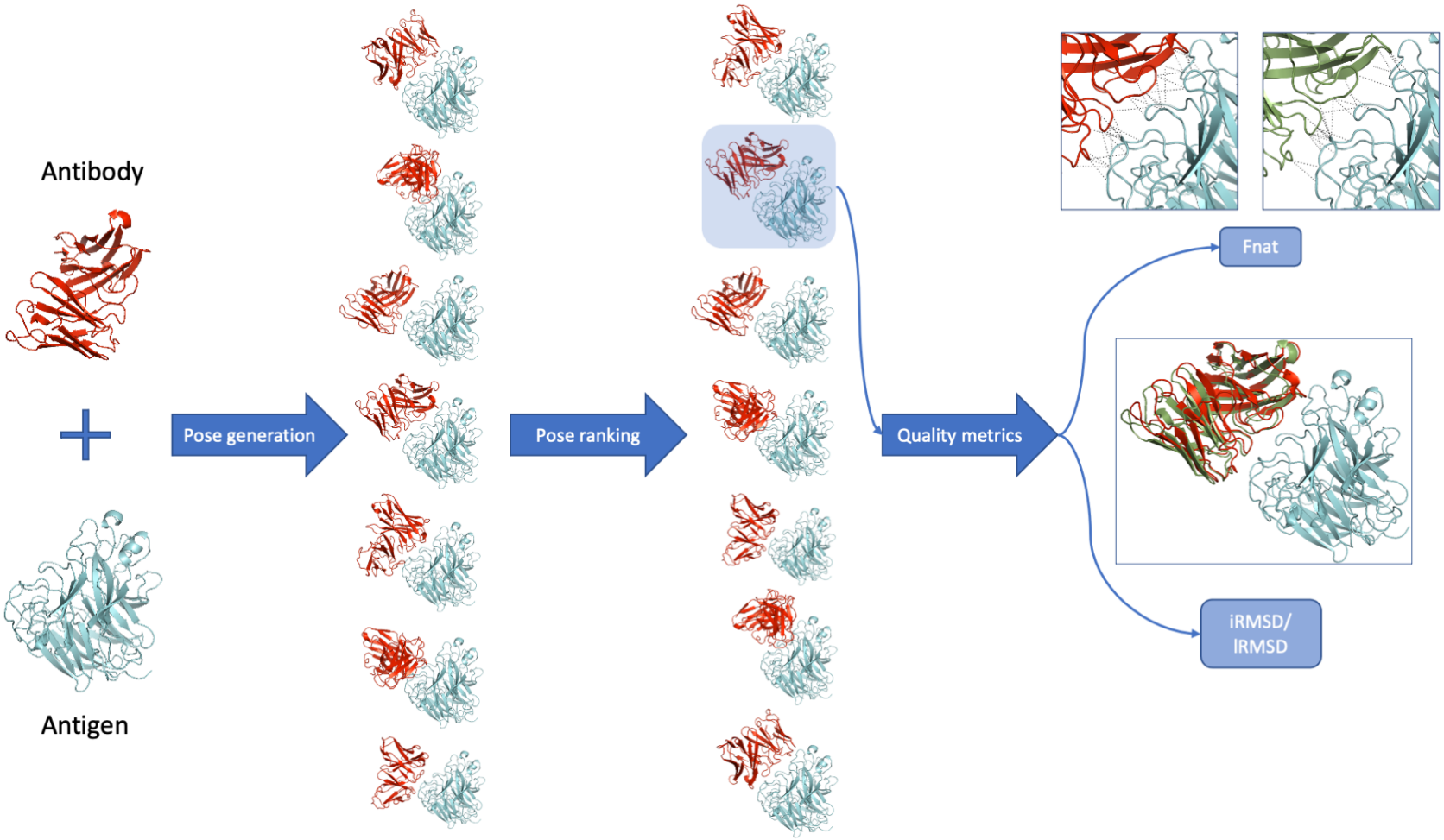

Figure 1.8: Schematic overview of the antibody-antigen docking process. The docking program generates 3D poses of the antibody and antigen. A scoring function then ranks those poses. If the true complex is known, the quality of the generated poses can be assessed via the fraction of native contacts reconstituted (*fnat*) or via RMSD calculations after aligning the true complex and the docked pose (iRMSD/lRMSD).

and are ranked by those scores (see Fig. 1.8). Some docking programs apply additional clustering to group similar poses and/or additional refinement of the high ranking poses at the interaction interface (Norman et al. 2020). In the following, I contrast different approaches to both protein-protein docking and specifically to antibody-antigen docking.

1.3.2 Approaches to protein-protein docking

There are two main approaches to computationally generate protein-protein complexes: rigid body docking and flexible docking, which differ in the degree of flexibility afforded to protein backbones and side chains. In rigid body docking (e.g. Pierce et al. (2011)), the input structures of interaction partners remain unchanged, with the docking algorithm generating translation and rotation matrices describing the transformations to apply to the input structures to generate docking poses. Rigid body docking tools can generate binding poses quickly and tend to perform well in scenarios with limited induced fit (Desta et al. 2020). In flexible docking approaches (e.g. Weitzner et al. (2017)), the relative coordinates of the side chain and/or backbone atoms of each interaction partner are updated as part of the docking process (Porter et al. 2019). Including flexibility in the generation of docked poses vastly increases the computational resources required. Whereas the rigid body docking tool ZDOCK can complete a docking run in approximately ten minutes (Pierce et al. 2011) on a CPU in use in 2011[12], the flexible docking tool SnugDock (Sircar and Gray 2010) takes around 300 hours to produce an antibody-antigen pairing on a CPU in use in 2010. Therefore, docking programs often combine the two approaches (Kozakov et al. 2017) to reduce computation time while still allowing some

[12] On the CPUs used for the research detailed in this thesis, this was reduced to approximately five minutes.

flexibility in the pose generation process.

Protein-protein docking programs also differ in their scoring functions. There are two main types of scoring functions: physics-based and statistical. Physics-based scoring functions evaluate the docked poses using energy-based interaction terms. Statistical scoring potentials on the other hand evaluate docked poses based on statistical analysis of experimentally determined complex poses, for example deriving contact potentials for residue-residue contacts (Krawczyk et al. 2013). Machine learning based scoring functions can be considered statistical scoring functions. Docking tools often use scoring functions combining physics-based and statistical terms (Pierce et al. 2011).

Additionally, molecular dynamics (MD) simulations, in which the state of a multi-atom system is iteratively updated based on a simulated force field, have been used for the study of protein-protein complexes as well as for the improvement of poses generated by docking programs (Lazim et al. 2020). However, even coarse-grained MD simulations for protein-protein docking require significant computational resources for the elucidation of each complex (Krupa et al. 2021). The computational cost associated with the application of MD approaches to protein-protein complex prediction makes them unsuited for the high-throughput settings I explore in the research detailed in this thesis.

1.3.3 Antibody-antigen docking

Knowledge of the precise location of the interaction site on the binding partners can aid in the generation of docking poses, as it limits the space of potential docking conformations to be

explored by the algorithm (this is referred to as information-driven docking). In the antibody-antigen domain, where the paratope can be predicted with high accuracy (see section 1.6.2.2), significant increases in speed and precision can be gained by using this information (Ambrosetti et al. 2019; Ambrosetti et al. 2020), either through docking tools explicitly created for the antibody domain (Sircar and Gray 2010; Kozakov et al. 2017) or by limiting the docking interface of general protein-protein docking tools (Pierce et al. 2011; Torchala et al. 2013). Similarly, restricting the region on the antigen that can be used in the docking interface (i.e. marking the epitope) can improve antibody-antigen docking results (Ambrosetti et al. 2019). While epitope prediction is considerably harder than paratope prediction (see section 1.6.2.2), in a virtual screening setting a target epitope is often defined in advance. In the following, I provide a brief overview of docking tools commonly used for antibody-antigen docking.

ZDOCK (Pierce et al. 2011) is a fast rigid body docking tool. It samples conformations using Fast Fourier Transforms (FFT) and scores individual poses using shape complementarity, electrostatic interactions, and a pairwise atomic statistical potential.

SnugDock (Sircar and Gray 2010) is based on RosettaDock and provides antibody-specific flexible refinement of docked poses provided by rigid body docking tools. It has recently been updated (Jeliazkov et al. 2020) and has been shown to produce highly accurate interaction poses (Guest et al. 2021) but is computationally expensive.

HADDOCK (Dominguez et al. 2003) uses rigid body energy minimisation and flexible refinement while incorporating information on the putative interaction site on both binding part-

ners into the docking algorithm, making it well suited for antibody-antigen docking.

SwarmDock (Torchala et al. 2013) is a flexible docking tool jointly optimising global position and conformation parameters of a "swarm" of 350 initial parameter combinations.

CLUSPRO uses the FFT-based rigid body algorithm PIPER for pose sampling, followed by clustering of low-energy poses and energy minimisation. It has an antibody mode, which uses an antibody specific scoring potential (Kozakov et al. 2017).

While flexible docking can improve results for antibodies displaying large induced fit (Desta et al. 2020), their long run time makes them currently unsuited for virtual screening of large antibody libraries. Further, affinity maturation of antibodies (see Section 1.1.3) usually also results in increased rigidity of the antibody CDR loops (Li et al. 2015), meaning that while omitting flexibility during docking might decrease the accuracy of docked poses of antibodies which exhibit large flexibility in the CDR loops, this is likely to be less problematic for antibodies binding their cognate antigens with high affinity. For these reasons, I elected to employ only rigid-body docking in the research detailed in this thesis and specifically used ZDOCK to generate docked poses. Most rigid body docking tools for antibody-antigen docking produce very similar results (Guest et al. 2021; Desta et al. 2020), with ZDOCK amongst the best performing ones, while also being fast to run.

Over the last year, several machine learning based general protein docking tools trained on structure data were released, which are discussed in section 1.6.2.3. In chapter 3, I describe the development of DLAB-Re, a novel antibody-antigen specific docking pose rescoring tool

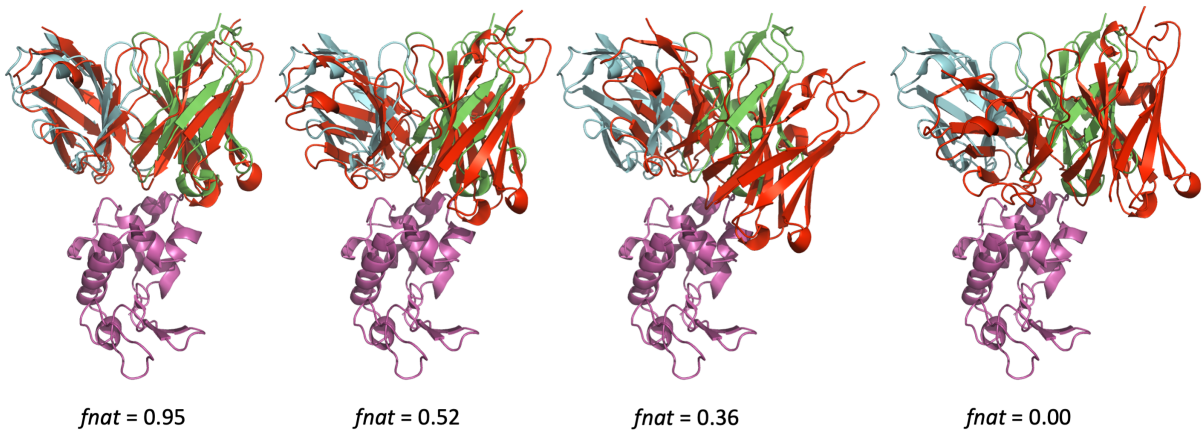

Figure 1.9: Comparison of docked poses of different quality. The chains of the original crystal structure are coloured blue (antibody light chain), green (antibody heavy chain) and purple (antigen). The docked antibody structure is coloured red. Four poses of decreasing quality from left to right are shown. The antibody and antigen crystal structures are taken from PDB 1fdl, the docked antibody was modelled using ABodyBuilder

learning directly from antibody-antigen interaction poses.

1.3.3.1 Assessment of docking performance

Similar to the CASP competition for protein structure modelling, there is a community experiment for docking algorithms: the Critical Assessment of PRedicted Interactions (CAPRI) (Wodak and Méndez 2004). The results of recent iterations of this experiment demonstrate that, as opposed to protein structure modelling, protein complex modelling is still an unsolved problem, both for general protein-protein contacts and for antibody-antigen contacts specifically. While high-quality poses may be produced during pose generation by various algorithms, scoring functions often fail to promote acceptable poses to the top of the pose list (Lensink et al. 2020; Lensink et al. 2021).

Figure 1.10: The drug discovery pipeline. The steps displayed are described in section 1.4.1. Steps coloured in blue or orange are accessible using *in-silico* techniques, steps coloured in green are *in-vivo* trials which by definition must be conducted *in-vivo*. Hit-ID (coloured in orange) is the step primarily addressed in this thesis.

In the CAPRI experiment, the quality of docking poses is assessed in reference to experimentally determined crystal structures using three metrics: interface root mean square deviation (iRMSD) (calculated from the distance of Cα atoms in the binding interface after aligning the true complex and the docked pose), ligand root mean square deviation (lRMSD) (calculated from the distance of C-α atoms in the mobile ligand after aligning the true complex and the docked pose) and *fnat*, the fraction of residue-residue contacts occurring in the experimentally determined structure which are also present in the docked pose (see Figs. 1.8 and 1.9).

In the research detailed in this thesis, I used either the full CAPRI criteria or *fnat* as a single regressor variable which directly captures the feature most important for downstream analysis and prediction of antibody-antigen interaction: the recapitulation of contacts.

1.4 The antibody drug discovery process

1.4.1 The drug discovery pipeline

Drug discovery campaigns, be it for small molecular drugs or biologicals, follow a similar pipeline (see Fig. 1.10). Initially, a disease target of interest is identified and validated. This

involves understanding disease pathways of interest and testing the effects of target inhibition on those pathways (Emmerich et al. 2021). Once a target (antigen) of interest has been identified, large scale screening experiments are used to identify initial hits, low affinity binders to the target (Gao et al. 2002). This step in antibody therapeutics discovery is the main focus of the research detailed in this thesis. Hits are then developed into lead compounds (hit-to-lead), increasing binding affinity and reducing properties undesirable for further development (Wan et al. 2021). Lead compounds are further developed, aiming to improve safety and efficacy profiles *in-vivo*, which includes animal models and cross-reactivity assays, before being passed into clinical trials, where candidate therapeutics are first tested in a small cohort of healthy volunteers (less than 100 participants) to assess safety (phase I), then in a larger cohort of patients to assess efficacy and immediate side effects (phase II) and lastly in an even larger cohort to further establish efficacy and assess long-term adverse reactions (phase III) (U.S. Food and Drug Administration 2018), before being approved and monitored post-approval (phase IV).

Drug discovery and development is costly. Estimates vary, with mean estimates for the cost to bring one drug to market (accounting for failed trials) ranging from $314 million to $2.8 billion (Wouters et al. 2020; Schlander et al. 2021). This cost is largely driven by failures in late stages of drug development: while 70% of candidate therapeutics move from phase I to II, this reduces to 33% for II to III and 25%-30% for successful phase III trials, meaning that only approximately 6% of therapeutics entered into clinical trials get approved. More accurate prediction of properties of hit and lead compounds could aid with the reduction of this failure rate.

The research detailed in this thesis focuses on pre-clinical antibody therapeutics discovery, which can be subdivided into several categories: *in-vivo*, *in-vitro* and *in-silico* (Laustsen et al. 2021; Sormanni et al. 2018).

1.4.2 *In-vivo* approaches to antibody drug discovery

In-vivo strategies for antibody discovery rely on exposing model animals to the target antigen and identifying and producing the antibodies that were raised against it (Laustsen et al. 2021). In the simplest form, this can be achieved via harvesting of the animal plasma and purifying antibodies as a polyclonal mix. Monoclonal antibodies can be generated from animal models after exposure to the antigen using several well-established methods (e.g. hybridoma (Alkan 2004)) and screened *in-vitro* (Engvall and Perlmann 1972; Hearty et al. 2012) to identify antibodies raised against the antigen target (see Fig. 1.11). Antibodies generated with these techniques can be cross-reactive and immunogenic, posing hurdles for downstream development (Bradbury and Plückthun 2015; Gray et al. 2016; Bradbury et al. 2018).

Large-scale sequencing has in recent years enabled the determination of large amounts of the antibody repertoire of immunised model animals or human patients and the use of those repertoires to generate antibody libraries for *in-vitro* screening (see section 1.4.3 and Raybould et al. (2021b), Kovaltsuk et al. (2018), and Marks and Deane (2020)).

Antibodies identified from animal models often have to be engineered in order to reduce immunogenicity (e.g. Marks et al. (2021)) and to avoid common developability problems (e.g. Raybould et al. (2019b)). Beyond humanisation to reduce immunogenicity (making the antibody

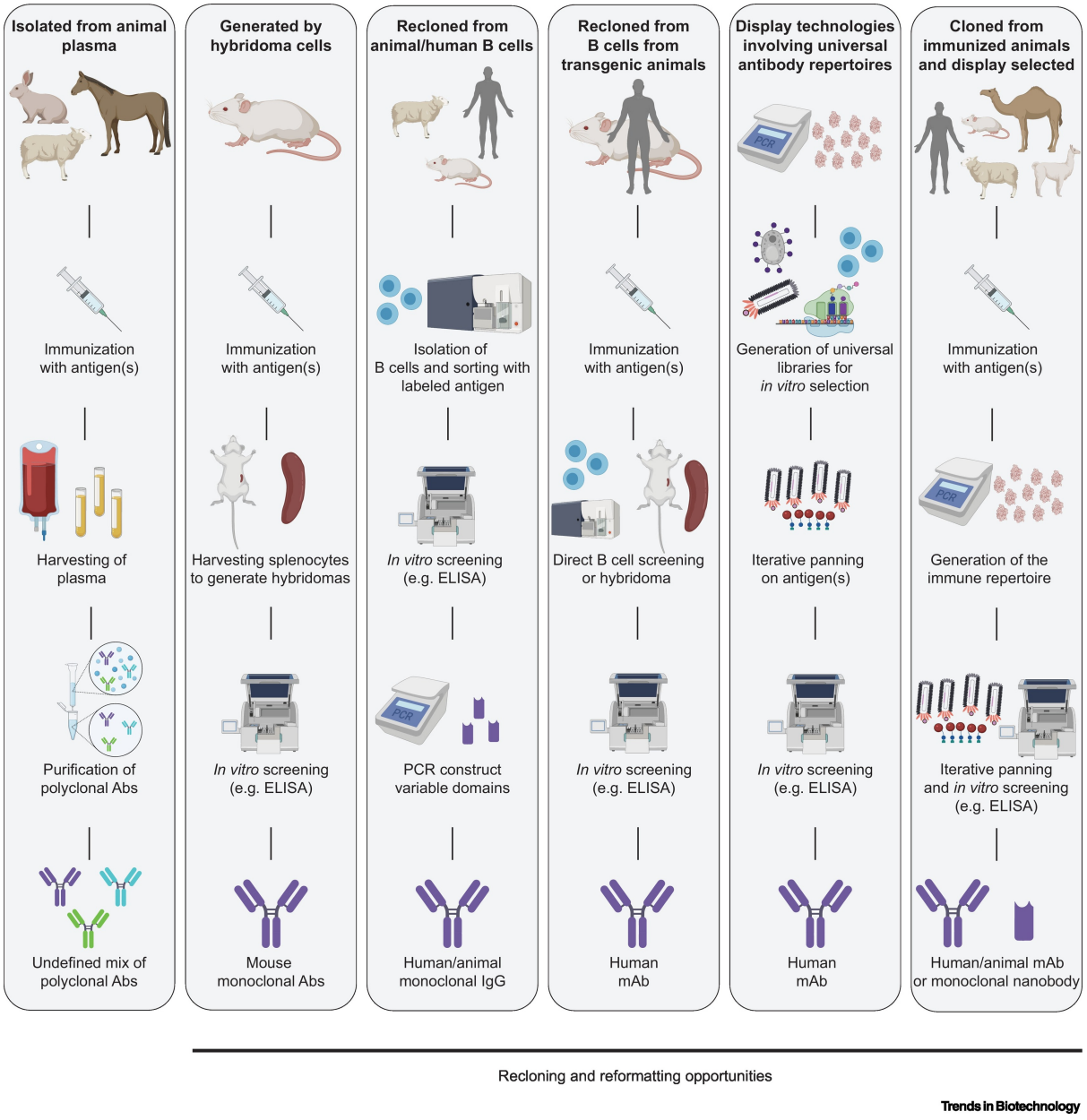

Figure 1.11: Schematic overview of approaches to antibody discovery. Figure reproduced from Laustsen et al. (2021) under the terms of the Creative Commons BY 4.0 license (https://creativecommons.org/licenses/by/4.0/).

more human-like, by either introducing mutations into the animal antibody (Safdari et al. 2013) or grafting animal Fv or CDR on human antibodies), recently transgenic rodent models have been used to generate human antibodies in animal models *in-vivo* (Teng et al. 2020; Lee et al. 2014).

1.4.3 *In-vitro* approaches to antibody drug discovery

In-vitro discovery approaches, as opposed to *in-vivo* approaches, do not use animal models to identify candidate antibodies for further engineering. Instead, for these approaches large libraries of antibodies are screened against the antigen target, primarily using phage display (Lee et al. 2014), where antibody heavy chains and light chains are combinatorically expressed on the surface of bacteriophages. Phage display libraries generated in this way tend to contain between 10^8 and 10^{10} distinct VH/VL pairings (Lee et al. 2014). These libraries are washed over plates of immobilised target antigen to enrich binding antibodies and binding antibodies are sequenced and further characterised and engineered. Antibody libraries generated for this purpose can be general, covering as much of the antibody sequence space as possible, or task/antigen-specific, for example by generating libraries from repertoires derived from immunised animals or human patients (Tiller et al. 2013; Kügler et al. 2018).

1.4.4 *In-silico* approaches to antibody drug discovery

In-vivo and *in-vitro* discovery approaches are cost- and time-intensive, while also requiring extensive know-how and equipment (Laustsen et al. 2021). Both are therefore supplemented extensively by computational, *in-silico* approaches (Sormanni et al. 2018).

In small molecule *in-silico* drug discovery, methods can be broadly separated into ligand-based (only considering the drug itself, for example using string representations) and structure-based (considering the three-dimensional interaction between the drug and its target) (Boyles et al. 2019). Correspondingly, approaches in antibody *in-silico* discovery can be subdivided either by the task within the pipeline they address or by whether they are only considering information on the antibody (usually the amino acid sequence) or information on the interaction structure between antibody and antigen.

In-silico approaches have been developed for the prediction of paratope and epitope residues (e.g. Liberis et al. (2018), Krawczyk et al. (2014), and Pittala and Bailey-Kellogg (2020), see also section 1.6.2.2). Beyond the value of this information for information driven docking (see section 1.3.3), successful paratope prediction can highlight residues on the antibody that should be targeted for affinity modulation, while accurate epitope prediction would be useful to confirm antibody binding to the active site. Structure- and sequence-based computational methods have been implemented for both tasks. For paratope prediction, sequence-based approaches and approaches considering only the antibody structure have been successful: a current state-of-the-art paratope predictor, Parapred (Liberis et al. 2018), takes as input only the antibody amino acid sequence. Accurate epitope prediction on the other hand has so far proven elusive (Pittala and Bailey-Kellogg 2020). While some success has been achieved on predicting linear epitopes (Bahai et al. 2021), the majority of epitopes are conformational epitopes (sequentially distant protein regions that are structurally close), which are harder to predict (Krawczyk et al. 2013; Pittala and Bailey-Kellogg 2020). Given that single antigens can have a variety of distinct epitopes bound by different antibodies (Kunik and Ofran 2013), as is the case for the

SARS-CoV-2 spike protein (Robinson et al. 2021) or lysozyme (Dunbar et al. 2014), and further given that there are little unifying features distinguishing epitopes from non-epitope protein surfaces (Kunik and Ofran 2013; Kringelum et al. 2013), recent better performing approaches take both the antibody and the antigen structure into account for epitope prediction (Pittala and Bailey-Kellogg 2020; Dai and Bailey-Kellogg 2021) (see also section 1.6.2.2 for an overview of current machine learning approaches to paratope and epitope prediction).

Computational approaches have also been developed to predict the binding affinity of antibodies against targets of interest or for *de-novo* generation of binders. Recently, several machine learning virtual screening approaches that aim to predict antibody binding against specific targets based solely on the antibody sequence have been published (Mason et al. 2021; Kang et al. 2021; Liu et al. 2019). These methods require prior experimental exploration of the binding landscape of the target of interest. Mason et al. (2021) for example were able to generate novel binders against HER2 using an input set of 10^4 experimentally determined data points.

Structure-based computational tools enable the *de-novo* generation of antibodies against target epitopes by fitting Fv regions generated combinatorically (often taking inspiration from V(D)J recombination) against an epitope of interest and performing multiple rounds of mutation (*in-silico* affinity maturation) (Sormanni et al. 2018). Commonly used approaches include RosettaAntibodyDesign (Adolf-Bryfogle et al. 2018), OptMAVen (Chowdhury et al. 2018) and AbDesign (Lapidoth et al. 2015), which all follow this general outline and have been experimentally validated. Programs following this outline can be used in conjunction with experimental screening (which in that case would provide initial hit antibodies) (Adolf-Bryfogle et al.

2018).

As mentioned in section 1.4.2, a major hurdle in antibody discovery beyond identifying high affinity antibodies for a target epitope is avoiding common developability problems. Recently, computational tools for the prediction of common developability issues as well as likely sources of immunogenicity have been released (e.g. Raybould et al. (2019b) and Chen et al. (2020)). The Therapeutic Antibody Profiler (TAP) (Raybould et al. 2019b) uses the structures of all therapeutic antibodies to derive a rules-based developability issue predictor. Hu-mAb (Marks et al. 2021) uses random forest classifiers to assign humanness scores to antibody sequences and suggest mutations to improve the humanness of those sequences.

While for small molecule drug discovery, virtual screening during the hit finding stage is commonplace (Maia et al. 2020), *in-silico* antibody discovery approaches primarily focus on improving the affinity of hits (Adolf-Bryfogle et al. 2018), finding novel binders for targets with large amounts of binding data available (Mason et al. 2021) or designing novel antibodies against epitopes of interest from the ground up (Chowdhury et al. 2018). A recent focus of work in the Oxford Protein Informatics Group, where I completed the research detailed in this thesis, based on research by Matthew Raybould (Raybould 2020; Raybould et al. 2019b), has been the development of a comprehensive in silico pipeline for the identification of candidate therapeutic antibodies from repertoire data using analysis on both sequence and structure level (see Fig. 1.12). For this suggested approach, antibody model libraries (AMLs) would be created from repertoire sequencing data (Raybould et al. 2021b) while pre-screening for developability issues (Raybould et al. 2019b; Marks et al. 2021). The representative antibody models of the

AML would be docked against a (computationally or experimentally defined) target epitope of interest and classified according to their binding potential against the target. Virtual screening tools like the DLAB tools I detail in chapters 3, 4 and 5 could enable the creation of this hit finding pipeline as detailed in Fig. 1.12.

1.5 Machine learning

1.5.1 A brief introduction to machine learning and deep learning

As defined by Mitchell (1997), machine learning describes a family of algorithms able to improve their performance on a well specified task with experience. Machine learning algorithms can be broadly divided into supervised, unsupervised and reinforcement learning approaches (Goodfellow et al. 2016), though there is overlap between these categories. Supervised algorithms learn from a set of labelled data points, aiming to predict the labels of unseen data points. Unsupervised algorithms learn from unlabelled input data and aim to find structure within the data (most clustering algorithms fall within this category). Lastly, reinforcement learning algorithms aim to find optimal policies to solve a given task while being provided with feedback in the form of reward functions depending on the actions chosen by the learning agents. In this thesis, I focus on supervised learning tasks.

Machine learning can further be divided into classical machine learning approaches and deep learning, where deep learning describes "representation learning approaches with multiple layers of representation" (LeCun et al. 2015). Deep learning approaches differ from classical

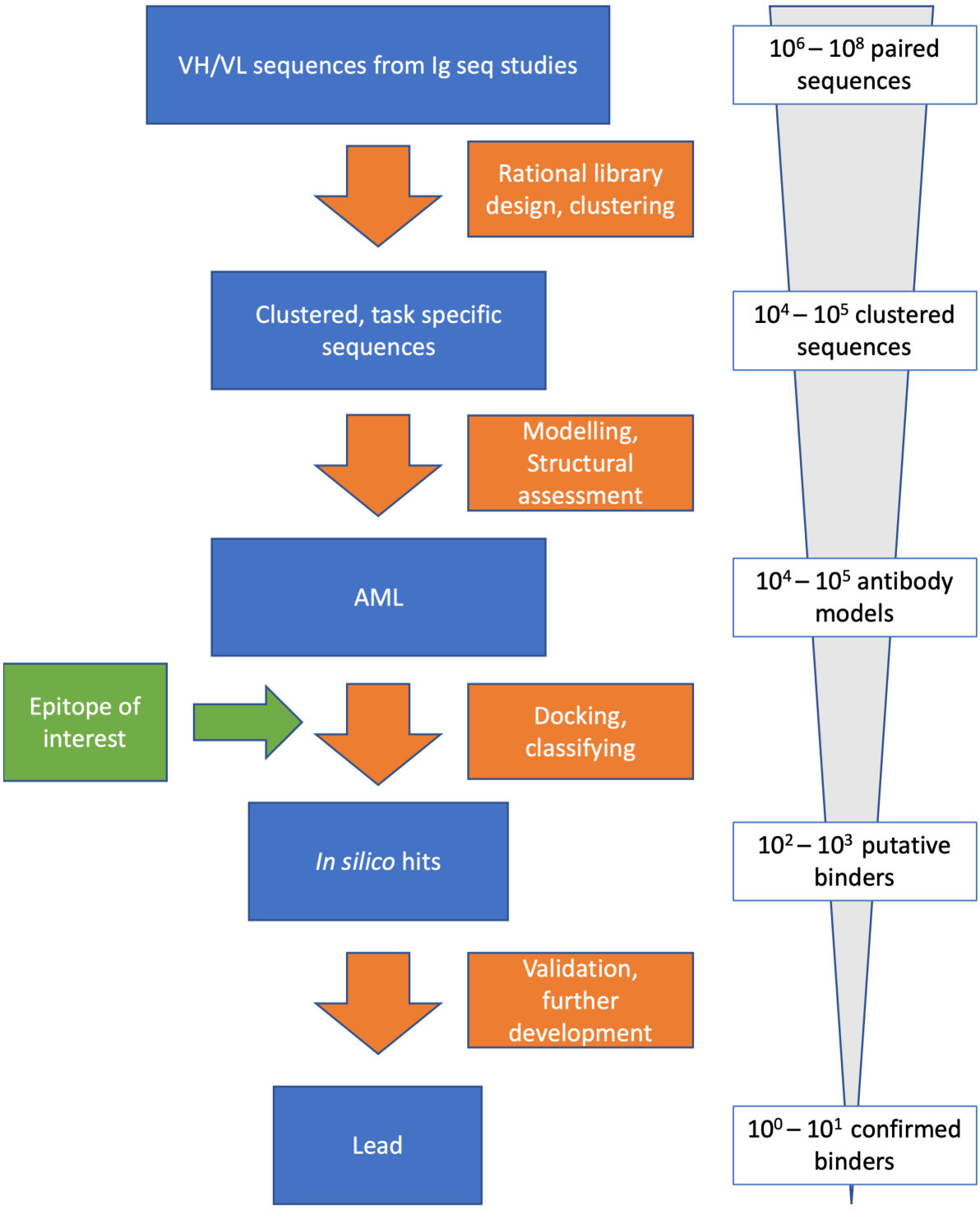

Figure 1.12: Proposed antibody discovery pipeline combining large-scale sequencing and structure-based virtual screening. Figure adapted from Raybould (2020).

machine learning in that they can effectively learn from raw input data, circumventing the need for feature engineering of the input data. The potential of deep learning algorithms has been realised in the last decade due at least in part to the advent of fast graphics processing units (GPUs), which enable them to be trained efficiently. As a result, deep learning approaches have become the gold standard in a wide variety of fields, from image recognition (Pak and Kim 2017), machine translation (Vaswani et al. 2017), and text generation (Brown et al. 2020) to protein structure prediction (Jumper et al. 2021).

Deep learning algorithms share common training principles (LeCun et al. 2015; Goodfellow et al. 2016). The training of deep learning algorithms relies on the differentiation of the trainable parameters of the model with respect to a metric quantifying the model performance, the loss function. This determination of parameter gradients is achieved through the backpropagation algorithm (Rumelhart et al. 1986). During training, model parameters are iteratively updated by changing their value by an amount defined by the model's learning rate and the parameter gradient with respect to the model loss. This procedure is referred to as gradient descent.

In the following, I will introduce the model types I use in the research detailed in this thesis: convolutional neural networks (CNNs) and graph neural networks (GNNs).

1.5.2 Convolutional neural networks

CNNs are machine learning algorithms designed to learn from spatially ordered data, using local connectivity and shared weights (LeCun et al. 2015). Their fundamental building blocks are N-dimensional filters/kernels with learnable weights, which perform convolution operations

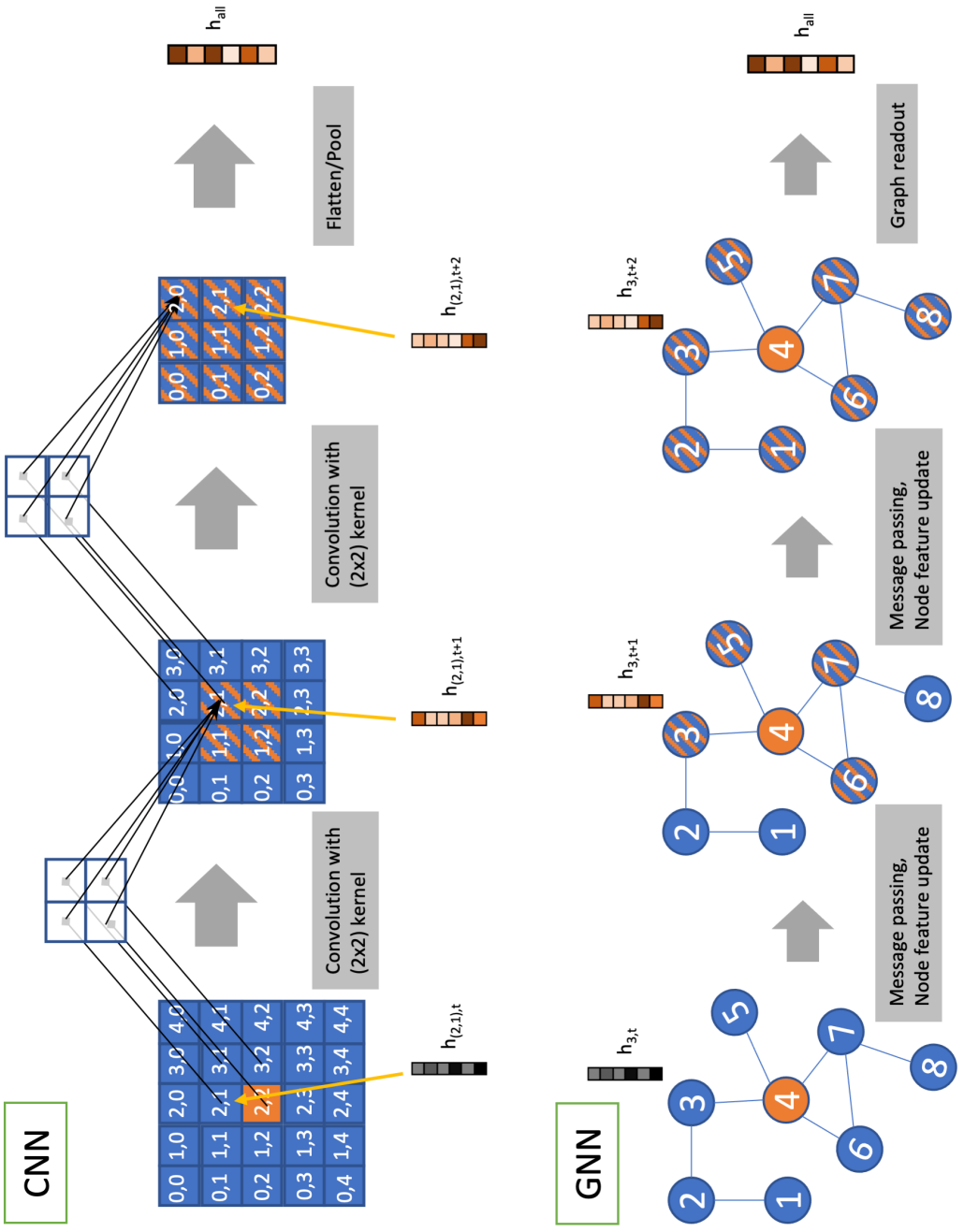

Figure 1.13: Comparison of information flow through CNNs and GNNs. In both cases, the information flow is visualised by showing, in orange, the distance over which information from one voxel (2,2) or node (4) can be aggregated. **(Top)** CNN: Kernels (in this example: 2x2, shown in white) convolve over the input data, combining local information using learnable weights and biases. To better show information flow, padding has been omitted in this example. **(Bottom)** GNN: Message passing enables updating of nodes based on the hidden features of their neighbours.

on the input data, which is provided as a (N+1)-dimensional array, according to the following equation[13] (see also Fig. 1.13):

$$h_{x,y,i,t+1} = \sum_{j=1}^{d} \sum_{n=0}^{k-1} \sum_{m=0}^{k-1} w_{n,m} * h_{x+n,y+m,t} + b \tag{1.1}$$

where $h_{x,y,i,t}$ is the hidden feature at position (x, y) in feature map i after layer t with kernel size k and d feature maps. $w_{n,m}$ and b are the weights and bias of the kernel respectively.

Individually, these kernels learn local features of the input data. By stacking layers of convolution operations, kernels in later layers have a larger effective receptive field. The more layers stacked in the CNN (i.e. the deeper it is), the more complex and long-distance patterns that can be learned by the CNN (Krizhevsky et al. 2012). CNNs further contain three additional layer types. Pooling layers aggregate information in adjacent positions (by taking either the mean or the maximum of adjacent positions) and thus make CNNs more robust against small distortions in the input data (LeCun et al. 2015). Regularisation layers like dropout (Srivastava et al. 2014), which randomly drops part of the network during training, reduce the tendency of neural networks to overfit to the training data (to memorise the labels of the training data set without generalising to unseen data). Lastly, non-linear activation functions, most commonly the ReLu function (Nair and Hinton 2010), introduce non-linearities into the CNN, which are essential in enabling the model to learn complex patterns, as otherwise any learned representation would reduce to a linear combination of the input features.

CNNs have been used for image analysis since the 1990s, with the LeNet architecture by Le-

[13] This equation describes the 2D case, the equivalent is true for higher (or lower) dimensional convolutions.

cun et al. (1998) being the most notable example. The introduction and performance of the deep AlexNet architecture (Krizhevsky et al. 2012) in the 2012 ImageNet image classification competition first demonstrated that deep CNNs could outperform other approaches at classifying spatially ordered multidimensional data. Over the following years, several improvements to the AlexNet architecture were made, including tuning the depth of the network and layer parameters (Simonyan and Zisserman 2015; Szegedy et al. 2014). Further, several architectures changed the overall connectivity of the CNNs. Previously, CNNs were connected in a linear fashion (each layer being directly connected only to the preceding and the following layer). ResNets (He et al. 2016) introduced skip connections, which skip intermediary layers and enable later layers to more directly access the input data to the network. This concept was expanded on by Huang et al. (2016), who introduced DenseBlocks, in which each layer is connected to all preceding layers within the same block.

Following the success of CNNs on image classification benchmarks, they have found widespread adoption in the analysis of biological data, including biomedical image analysis (McQuin et al. 2018) and small molecule virtual screening (Imrie et al. 2018). In chapters 3 and 4, I explore the use of CNNs for the generation of structure-based antibody virtual screening tools.

1.5.3 Graph neural networks

1.5.3.1 Introduction

Over the last two years, deep learning approaches for predicting protein properties from protein graph representations have gained popularity, motivated partially by the success of AlphaFold2

(Jumper et al. 2021). These methods employ Graph Neural Networks (GNNs) to make predictions based on protein structural features.

A graph is a collection of nodes (also referred to as vertices) $V = \{v_1, v_2, ..., v_i\}$ with associated features $X = \{x_1, x_2, ..., x_i\}$ connected by edges $E = \{e_{0,0}, e_{0,1}, ..., e_{i,j}\}$, which themselves can have edge features. Edges can either be unidirectional or bidirectional, resulting in directed or undirected graphs. In the research detailed in this thesis, I solely consider undirected graphs. With an edge connecting node v_i to node v_j denoted as $e_{i,j}$, the neighbourhood N of node v_i is defined as $N(v_i) = \{v_j \in V | e_{i,j} \in E\}$.

Molecular structural data can be easily converted into graphs by considering atoms or residues to be nodes in the graph and constructing edges based on physical proximity (e.g. Satorras et al. (2021)) or on physical bond structure (e.g. Imrie et al. (2020)).

(Molecular) graphs are accessible to deep learning approaches with GNNs, which were first introduced by Scarselli et al. (2009). These pioneering models aimed to arrive at an equilibrium state for an input graph by recurrent application of the message passing algorithm until convergence. Message passing updates the features of each node based on its own representation and the representations of nodes in its neighbourhood according to the following equation:

$$h_v^{(t)} = \sum_{u \in N(v)} f(x_v, x_{(v,u)}^e, x_u, h_u^{t-1}) \tag{1.2}$$

where f is a learnable function which is required to fulfil a set of convergence criteria (Wu et al. 2021). This was later extended by Gilmer et al. (2017) to be equivalent to CNN architectures

by considering message passing a convolutional layer over the node neighbourhood according to the following equations (see also Fig. 1.13), rather than applying the same layer recurrently until convergence:

$$m_v^{t+1} = \sum_{u \in N(v)} M_t(h_v^t, h_u^t, x_{(v,u)}^e) \tag{1.3}$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) \tag{1.4}$$

where M and U are layer specific learnable functions, m_v^{t+1} is the sum of messages passed to node v, h_v^{t+1} is the representation of node v after layer t and $h_v^0 = x_v$. These layers can be stacked in the same manner as convolutional layers in CNNs and form the basis for many current state-of-the-art GNN implementations, including the EGNN (Satorras et al. 2021) which I use in chapter 5 to develop a graph-based antibody virtual screening tool.

GNNs can be applied to node or edge level classification, in which case additional fully connected layers are usually applied at the node/edge level to arrive at classification output, or to graph classification. In the latter case, which I employ in this thesis, node features need to be combined into a single vector for the graph, referred to as the graph readout (Zhou et al. 2020), which can be achieved through averaging or max-pooling of node features. For comprehensive reviews of GNNs, including non-message-passing approaches to information flow, see Wu et al. (2021) and Zhou et al. (2020).

1.5.3.2 SE(3) equivariant and invariant GNNs

Over the last few years, there has been rising interest in equivariant GNNs, a family of GNNs which are equivariant to certain groups. A function f is equivariant with respect to a specific group G (for example SE(3), the group describing translational and rotational symmetry operations in Euclidean space, excluding reflections) if

$$f(g \cdot x) = g \cdot f(x) \forall g \in G, \tag{1.5}$$

and invariant if

$$f(g \cdot x) = f(x) \forall g \in G. \tag{1.6}$$

All message-passing GNNs are invariant to permutations since message passing aggregation is order-agnostic. Similarly, if no three-dimensional coordinates are supplied as part of the graph features, general GNNs are rotation- and translation-invariant (SE(3)-invariant) since no information about Euclidean geometry is preserved in the model. For machine learning models operating on molecular data, SE(3) equivariance is a desirable property, as it captures the inherent three-dimensional nature of molecular structures, avoiding the need for extensive data augmentation during training. Therefore, driven by new GNN architectures implementing equivariant models (Fuchs et al. 2020a; Satorras et al. 2021; Hutchinson et al. 2021) and the success achieved by AlphaFold2 using the equivariant structure module (Jumper et al. 2021), a large number of studies using equivariant GNNs on molecular data have been released in the

past two years (Abanades et al. 2021; Eismann et al. 2020a; Eismann et al. 2020b; Fuchs et al. 2020a; Satorras et al. 2021; Hutchinson et al. 2021; Jing et al. 2020; Jing et al. 2021).

There are currently three main approaches to implementing equivariant graph networks. There are several publications (based on the tensor field network published by Thomas et al. (2018)) using spherical harmonics to perform equivariant graph attention[14] or convolutions (Fuchs et al. 2020a; Eismann et al. 2020a; Eismann et al. 2020b). These approaches come with the caveat that they require expensive computations during each forward pass through the model, making them both slow and memory-intensive (Fuchs et al. 2020a). A similar method, also using group theory approaches to perform equivariant convolutions or attention was published by Finzi et al. (2020) and Hutchinson et al. (2021), using lifting as opposed to spherical harmonics. Lastly, several studies have been published implementing an equivariant version of the message passing equations laid out above, enabling the equivariant incorporation of vector features into message passing networks (Jing et al. 2020; Jing et al. 2021; Satorras et al. 2021). These methods achieve equivariance through inclusion of the L2-norm of vector features in the message passing algorithm and are thus more computationally efficient than the two former approaches.

Due to their comparative ease of implementation and superior speed, I primarily used the latter implementation in the research detailed in chapter 5.

[14] Attention here describes a machine learning algorithm which gained popularity first in language processing (Vaswani et al. 2017) and was later extended to other fields, including protein modelling (Jumper et al. 2021). With attention, all inputs for a given computation (messages in the context of graph attention) are assigned a score. These scores are passed through a softmax layer and used to weight the corresponding inputs, thus enabling the network to learn how much "attention" to pay to its inputs at any given point.

1.6 Machine learning approaches for antibody therapeutics discovery

Machine learning algorithms are commonly used in *in-silico* drug discovery, including for antibody discovery (Graves et al. 2020). In this section, I highlight machine learning applications relevant to the structural analysis of antibody-antigen interactions performed for this thesis.

1.6.1 Relevant application in small molecule drug activity prediction

Machine learning-based approaches have long been established in small molecule drug discovery, see for example the reviews by Vamathevan et al. (2019) and Lo et al. (2018). Here, I highlight studies which have influenced the research detailed in this thesis. Ragoza et al. (2017) developed a CNN-based small molecule binding classification model using docked protein-ligand interaction poses derived from the CSAR and DUD-E benchmarking data sets as input. The method outperformed all other scoring functions at publication. The poses were discretised into a three-dimensional grid of atom type channels using an atom type density description based on atom radii, which was published as a standalone python package (Sunseri and Koes 2019). This work was later extended upon by Imrie et al. (2018) using state-of-the-art DenseNet architectures to outperform the original architecture and Scantlebury et al. (2020), demonstrating important considerations surrounding pose quality in the training data set. Much of the work in chapters 3 and 4 is based on the models detailed in those studies.

Other models using similar atom type gridding approaches were published concurrently for

protein-ligand binding site prediction (Jiménez et al. 2017) and protein-ligand affinity prediction (Jiménez et al. 2018).

1.6.2 Applications of machine learning to protein-protein and antibody-antigen interaction

For protein-ligand interaction, large, standardised data sets exist which can be drawn upon for machine learning model training (Gaulton et al. 2012; Bauer et al. 2013; Mysinger et al. 2012). While not without issues (Chen et al. 2019), these data sets enable the quick creation of comparable models.

For the structural analysis of protein-protein interaction (PPI), fewer standardised data sets exist (though there are recent attempts at creating such benchmarking data sets (Townshend et al. 2019; Townshend et al. 2020; Morehead et al. 2021)[15], meaning that training sets in this field tend to be curated for specific studies. For data sets used to train machine learning models of antibody-antigen interaction, care has to be taken during the creation of data splits to avoid the inclusion of too similar paratope-epitope interfaces in data sets used for training and validation/testing (see chapter 4).

In the following, I present recent machine learning approaches to address tasks relevant for the analysis of antibody-antigen interaction. Antibody-antigen interaction (ignoring non-protein antigens) can be considered a subset of PPI, meaning that advances in the prediction of properties of PPIs described below are applicable to antibody-antigen interaction as well. I detailed

[15]Note that DIPS (the database of interacting protein structures) (Townshend et al. 2019) is distinct from DIP (the database of interacting proteins) (Xenarios 2002).

recent advances in general protein and antibody modelling, including using structure-based deep learning, above (see section 1.2.2) and will therefore omit this task in the following section. I further limit this section mostly to structure-based deep learning approaches. There is a growing body of literature on language models applied to amino acid sequences (e.g. Kang et al. (2021), Saka et al. (2021), and Brandes et al. (2021), which, though beyond the scope of this thesis, may well be a future direction of this type of work.

1.6.2.1 Protein-protein interaction site prediction

Recently, two structure-based geometrical machine learning approaches have achieved state-of-the-art results on predicting the site of interaction between two proteins. Rather than using graph-representations, both of those use geometric representations of the protein surface. MaSIF (Gainza et al. 2020) defines patches on the protein surface with fixed radius and represents features of the patches using polar coordinates and uses geodesic convolutions over the polar features to derive fingerprints, training the network to minimise the distance between the fingerprints of interacting patches. PInet (Dai and Bailey-Kellogg 2021) uses a similar approach, defining the surface of interaction partners as a parameterised point cloud which is transformed to be invariant to SE(3), calculating both local and global features of each protein using multi-layer perceptrons (MLPs) to jointly predict interface segments on both structures.

1.6.2.2 Antibody-antigen interaction site prediction: paratope and epitope

As stated in section 1.4.4, most paratope prediction tools use information from the antibody alone (e.g. Parapred (Liberis et al. 2018)). Several recently released tools employ deep learn-

ing approaches learning jointly from paratope and epitope structure to achieve state-of-the-art paratope prediction and, importantly, improve epitope prediction.

Pittala and Bailey-Kellogg (2020) created PECAN, which predicts epitope and paratope residues from graphs with residue-level nodes and distance-defined edges of both antibody and antigen, parameterising each node with residue type as well as sequence conservation, solvent accessible surface, and local amino acid usage environment. Their network performs graph convolutions on both antibody and antigen and derives a spatial readout from the graph for which the prediction is performed (i.e. the antibody graph for the paratope, the antigen graph for the epitope) and a context vector via attention over both graphs. Other than PInet, the PECAN architecture is asymmetric and can therefore not be trained jointly, requiring a separate network to be trained for paratope and epitope prediction.

Del Vecchio et al. (2021) implemented two message passing networks for paratope (Para-EPMP) and joint paratope/epitope (Epi-EPMP) prediction. Their architectures have two interesting features. Para-EPMP combines antibody sequence and structure to make paratope predictions, first generating an embedding for each residue using one-dimensional convolutions and then adding those embeddings to a graph representation, on which several message-passing layers are applied. Epi-EPMP combines the attention-based context generation seen in PECAN with joint prediction of paratope and epitope seen in PInet.

Lastly, a version of PInet (see above) has been trained specifically for joint paratope-epitope prediction, achieving improved performance on the epitope prediction task over previous tools as well as PECAN and EPMP (Dai and Bailey-Kellogg 2021).

1.6.2.3 Protein-protein complex structure prediction

Machine learning approaches for complex structure prediction can be divided into two approaches: They either assess the quality of generated complexes, serving as scoring functions for better ranking of docked poses, or they directly generate a prediction for the complex structure.

In the former category, several structure-based machine learning approaches have enabled the direct assessment of complex pose quality from structures without additional feature engineering. DOVE (Wang et al. 2019b) uses CNNs, creating discretised grids from docked structures. It uses four atom types alongside contact potentials to generate the input grids and predicts a classification score to indicate if an input pose meets the CAPRI acceptability criteria. In a recent update to DOVE architecture, the authors switched to a GNN based implementation, outperforming their previous method (Wang et al. 2021a). A similar trend could be observed in two publications released in December 2021: DeepRank, a CNN-based docking pose evaluation model (Renaud et al. 2021) was improved upon by switching to a GNN-based implementation (Réau et al. 2021). Yasser et al. (2021) implemented a locally invariant CNN approach for docking pose evaluation by partitioning the interface into cubes which, rather than being oriented in the same coordinate system, are oriented along the protein backbone. Eismann et al. (2020a) used an equivariant graph architecture to predict pose lRMSD. Their network circumvents some of the challenges arising from large protein graphs with atom-level nodes by introducing a hierarchy of graph-levels: The first layer sees all atoms in the structure, while the graph in following layers is constructed from Cα atoms. Their method, PAUL, was shown to

improve docking pose ranking.

In the latter category, AlphaFold2 has been adapted for the prediction of multimeric proteins (Evans et al. 2021) and for the prediction of heterodimeric protein complexes (Yin et al. 2021; Bryant et al. 2021) and has been combined with a classical rigid body docking tool (Ghani et al. 2021). Interestingly, while these extensions of AlphaFold2 perform well on general protein complexes, like the original AlphaFold2 model, they perform less well in the antibody domain. Ganea et al. (2021) recently suggested an end-to-end trainable rigid body docking approach, termed EquiDock, based on an extension to the equivariant architecture suggested by Satorras et al. (2021), which rather than predicting complexes from sequences, predicts the transformations to be applied to input files. While not yet achieving state-of-the-art performance compared to other rigid body docking algorithms, this approach is intriguing as it removes the need for computationally expensive sampling of the conformational space, with CPU run times of 5s per complex.

The approaches above have been trained primarily on experimentally determined structures and are not specific to the antibody domain. In chapter 3, I describe DLAB-Re, an antibody-specific, CNN-based docking quality assessment and rescoring tool trained to specifically assess the docked poses of computationally generated antibody models.

1.6.2.4 Structure-based antibody-antigen binding prediction and virtual screening

There have been comparatively few studies assessing the potential of structure-based machine learning approaches for antibody binding prediction, design, and virtual screening.

Ripoll et al. (2021) used two-dimensional projections of computationally modelled paratope structures to (amongst other tasks) distinguish binders against different epitopes on the same antigen but did not extend their method to more general binding classification.

Jespersen et al. (2019) partitioned the surface of experimentally determined antibody and antigen structures into distinct patches, which are described by their amino acid composition, several features describing the chemical and physical composition of the patch as well as several features describing the patch geometry. These patch features were used to train a feed-forward neural network to discriminate between binding and non-binding patch pairings.

Akbar et al. (2021a) extracted one-dimensional interaction motifs from the paratope and epitope of experimentally determined antibody-antigen complex structures and used these to train a machine translation model to translate between paratope and epitope motifs, demonstrating learnability of the structural profile of the paratope-epitope interaction.

As the low volume of available structural data on antibody-antigen interaction (Dunbar et al. 2014) can limit the expressiveness of machine learning models trained on real-world data, Robert et al. (2021) created Absolut!, a software suite to generate synthetic lattice-based antibody-antigen complex structures. Their software enables the rapid generation of large data sets with similar properties to the structural space of antibody-antigen interaction and can be used to demonstrate feasibility of different structure-based machine learning approaches for the antibody-antigen domain and for benchmarking of algorithms in a high-data environment. Akbar et al. (2021b) used this software to demonstrate the feasibility of a sequence-based transfer learning strategy for binder generation.

As this thesis was written, CSM-AB was published (Myung et al. 2021), which uses graph-based signatures of the antibody-antigen interaction site as input to several machine learning model types in order to both predict binding affinity and serve as a docking scoring function.

In chapters 4 and 5, I describe the development of two antibody-antigen binder classification tools trained specifically to predict binding from computationally generated antibody structures.

1.7 Thesis outline

In this chapter, I have laid out the biological background of antibody-antigen binding as well as the usefulness of antibodies as therapeutics and the need for effective high-throughput virtual screening tools to supplement antibody discovery.

In chapter 2, I detail my contributions to the curation of SAbDab, the Structural Antibody Database, which is used throughput this thesis for the creation of bespoke data sets. These include the implementation of advanced search tools and the native integration of ancillary databases into SAbDab.

In chapter 3, I describe the development of DLAB-Re, a CNN-based model for the rescoring of antibody docking poses and the assessment of docking quality which is trained specifically on computationally generated antibody models.

In chapter 4, I extend the method developed in chapter 3 to binder/non-binder classification of antibody-antigen pairings and provide a proof-of-principle for the feasibility of structure-based

machine learning models for fast high-throughput virtual screening of antibody model libraries.

In chapter 5, I revisit the binder/non-binder classification task through the lens of the novel model class of equivariant graph neural networks, demonstrating that these can achieve a strong improvement over CNN-based models for the classification of antibody-antigen pairings.

In chapter 6, I summarise the contributions made by the research described in this thesis and provide suggestions for the next steps in the development of effective structure-based virtual screening tools for antibody therapeutics discovery and give thoughts on the future direction of the field.

2 | Curation of antibody and nanobody structures using the Structural Antibody Database

Contents

2.1 Chapter motivation

In this thesis, I have used antibody and antigen structural data for the development of antibody-antigen virtual screening algorithms. Access to a well annotated and curated repository of antibody structures was therefore a fundamental requirement for my work as well as for future progress of similar algorithms. This chapter describes my work in maintaining and improving

the Structural Antibody Database (SAbDab), including the curation of SAbDab-nano, a sub-database that tracks nanobodies (heavy chain-only antibodies) which have seen a particular growth in attention from both the academic and pharmaceutical research communities over the past years (see section 1.1.5.2).

The contributions described in this chapter have been published as "SAbDab in the age of biotherapeutics: updates including SAbDab-nano, the nanobody structure tracker" (Schneider et al. 2021a), parts of this chapter are adapted from the paper.

2.2 Clarification of contribution

SAbDab, the database described in this chapter, represents a large body of work by multiple researchers over eight years. Beyond the original publication by Dunbar et al. (2014), Matthew Raybould, Claire Marks, Jinwoo Leem and myself have made significant contributions to SAbDab. In the following text, I state which parts of the current database I contributed.

2.3 Introduction

Due to the importance of an accurate understanding of the three-dimensional structure of antibodies for the study of their properties and the development of antibody therapeutics, Dunbar et al. (2014) released the Structural Antibody Database (SAbDab) in 2013, a comprehensive and continuously updated database of experimentally determined antibody structures. Since its publication, SAbDab has been used in numerous studies for the creation of tailored antibody data sets, including for the creation of training sets for antibody-specific machine learning mod-

els (e.g. Schneider et al. (2021b), Robinson et al. (2021), Hou et al. (2021), and Myung et al. (2021)). SAbDab has rapidly increased in size: from 1624 entries when it was published to 5557 as of 10/11/2021 (see Fig. 2.3).

SAbDab is updated weekly by detecting antibody structures in the Protein Data Bank (PDB) (Berman et al. 2000) and numbering as well as annotating these structures with relevant additional information: name, species, experimental method, resolution, amino acid sequence including the complementarity determining regions (CDRs), the heavy and light chain subgroup, antigen type, and, where available, antigen binding affinity value. For an overview of SAbDab, see Fig. 2.1 and Dunbar et al. (2014). Since SAbDab was initially released, the database back end has been updated (see Fig. 2.1 and Schneider et al. (2021a)), now allowing fast searching of the database via an SQL database at the back end and a Flask python application at the front end, enabling the database to cope with the rapid increase in available antibody structures.

Over the last few years, several auxiliary databases have been created which catalogue antibodies with attributes of interest, including up-to-date collections of World Health Organisation-recognised therapeutic antibodies (Thera-SAbDab) (Raybould et al. 2020) and coronavirus-binding antibodies (CoV-AbDab) (Raybould et al. 2021a). These databases primarily contain antibody sequences, with rising interest in the analysis of antibody structures based on their therapeutic properties or their ability to bind to specific epitopes (e.g. Raybould et al. (2019b) and Robinson et al. (2021)) indicating the need to link structure databases like SAbDab to these antigen- or application-specific databases.

In this chapter, I describe several updates to SAbDab I implemented during my DPhil research:

linking the database to antigen- and application-specific databases as well as several advanced search, data set creation and automation features.

Further, in recent years, nanobodies, heavy chain-only antibodies which were first identified in camelids, have emerged as an important class of immune molecule. This class of antibodies has several properties making them good therapeutic candidates: they have low molecular weight, enabling easier manufacturing and crossing of the blood-brain barrier (Yang and Shah 2020), as well as longer CDR3 loops than comparable antibodies, enabling the targeting of epitope topologies inaccessible to other antibodies (Holliger and Hudson 2005) (see section 1.1.5.2 for a more detailed explanation of nanobody properties). A large body of research has now been published on their properties and their potential as therapeutics (e.g. Yang and Shah (2020) and Jovčevska and Muyldermans (2020)), including for SARS-CoV-2 (Huo et al. 2020). As of August 2021, one nanobody therapeutic has been approved and six more are in clinical trials (Raybould et al. 2019a). Meanwhile, the number of experimentally determined nanobody structures is also growing rapidly (see Fig. 2.3). While SAbDab has always contained nanobody structures, these recent trends have motivated the creation of SAbDab-nano, a sub-database of SAbDab which is the first nanobody-specific, continuously updated, and annotated structure database.

In this chapter, I further describe the creation and content of SAbDab-nano.

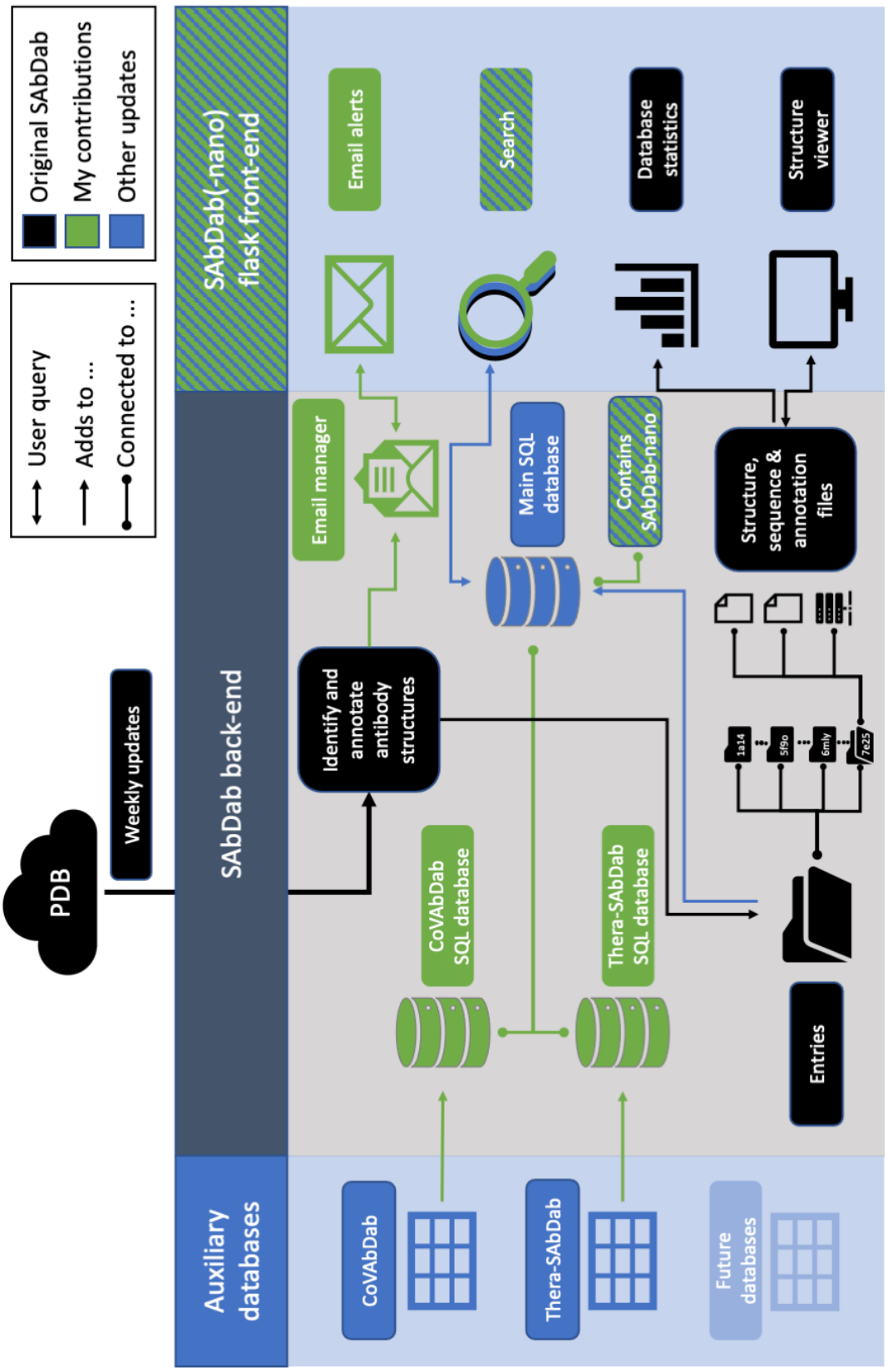

Figure 2.1: Overview of the SAbDab back and front end. The diagram is colour coded, black colouring highlights parts of the database mostly unchanged since inception and described in Dunbar et al. (2014), blue colouring highlights parts modified or added since inception, green colouring highlights my contributions to the database. Shaded fields refer to parts of the database to which I as well as other researchers have contributed.

2.4 Methods

2.4.1 Additional data annotation

As described above, there are currently two relevant auxiliary databases to SAbDab: the database of coronavirus-binding antibodies (CoV-AbDab), containing data on 5210 antibodies as of March 2022 and the database of World Health Organisation-recognised therapeutic antibodies (Thera-SAbDab), tracking 742 therapeutics as of March 2022. I linked both these databases to SAbDab using the following protocol.

Once per week, SAbDab is updated. This update step identifies new antibody structures in the PDB and annotates those structures. To enable fast searching over the database, after each update this data is compiled into a SQLite data table (the main SQL database in Fig. 2.1) using the sqlite3 python package.

For each auxiliary database, I added an additional SQL data table to SAbDab, which is filled after each update by identifying sequences in the two not SQL-based auxiliary databases which either have direct matches in SAbDab (CoV-AbDab) or which match an entry in SAbDab with at least 95% sequence ID (Thera-SAbDab) using the data annotation in those databases. These two auxiliary SQL tables are then populated with the relevant annotation from the auxiliary databases and indexed with the PDB code of the corresponding SAbDab entries.

Using these additional SQL tables, annotation derived from auxiliary databases can be accessed through the SAbDab structure viewer interface (see Fig. 2.2) and using the SQL-based

search interface, SAbDab can be filtered for entries with annotation derived from these auxiliary databases.

As of 10/11/2021, SAbDab contained 367 structures matching Thera-SAbDab entries, four of which were nanobodies and 435 structures matching CoV-AbDab entries, 71 of which were nanobodies.

2.4.2 Additional data access options

At publication, SAbDab contained many search features: searching by experimental method used to determine the structure, resolution and R-factor[1] cut-offs, species of the antibody, antigen type, presence of affinity values in the annotation and presence of amino acid residues at specific sequence positions defined using the Chothia numbering scheme. As part of my contribution, I have added several features to enable more specific searches and custom data set creation using SAbDab.

As mentioned above, I implemented the option to search specifically for structures with annotation derived from CoV-AbDab and Thera-SAbDab. I have also put in place a free-text keyword query over certain annotation fields (antigen, species, publication, and structure title), retrieving all structures for which the specified fields contain an exact match to one or more of up to ten key strings. Both of these new features improve the ability to create task-specific nanobody and antibody data sets. An example search query is displayed in Fig. 2.2A and B.

To help the community stay up to date with the release of antibody or nanobody structures of

[1]The R-factor measures the difference between the experimentally observed diffraction data and the computationally predicted diffraction pattern of a proposed protein structure.

interest, I also added a mechanism that generates email alerts based on bespoke user search queries. Using this interface, the search definition is saved on the database server alongside a user-provided email address. After each weekly update, new structures matching the search query are automatically identified and a summary email detailing those structures is sent to the provided address. This feature will, for example, enable users to be immediately notified when a structure of an antibody against an antigen target of interest is added to SAbDab.

2.4.3 SAbDab-nano

One of the key updates to SAbDab was the development of SAbDab-nano in order to improve the availability and accessibility of nanobody structures within SAbDab. Development of this new sub-database of SAbDab was started by Claire Marks and completed by me.

From the set of antibodies identified for addition to SAbDab in each weekly update, entries for which at least one antibody has a heavy chain variable domain, but no light chain variable domain, are added to SAbDab-nano and are made searchable and accessible using the same search and access features described above for the main SAbDab database (see Fig. 2.2).

2.4.4 Analysis of database contents

I calculated summary statistics for SAbDab and SAbDab-nano using the SAbDab python API, based on a snapshot of SAbDab taken on 09/09/2021. Where numbers for SAbDab are quoted, unless otherwise stated these include SAbDab-nano as a sub-database of SAbDab.

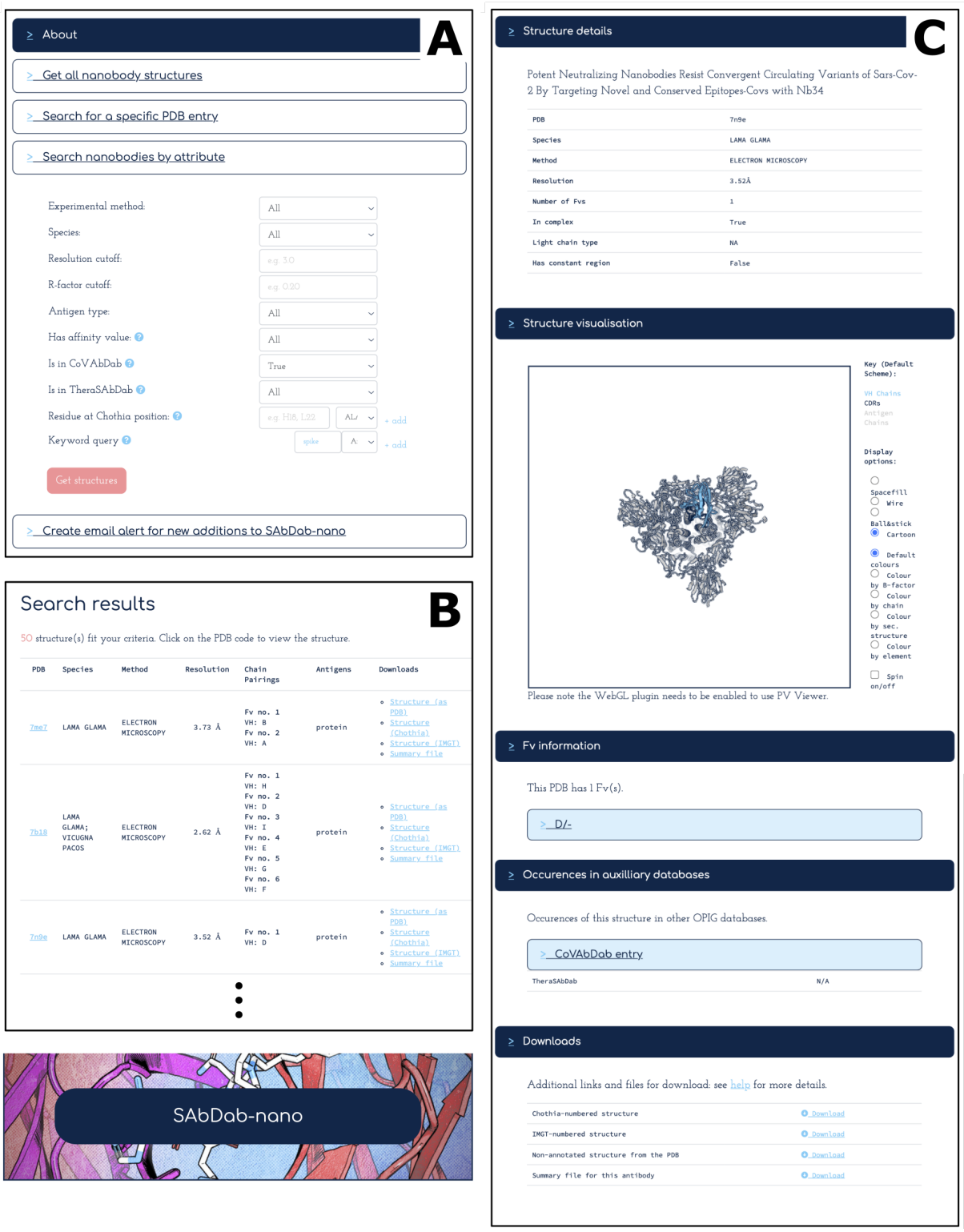

Figure 2.2: Overview of the SAbDab-nano search interface. Caption on next page.

Figure 2.2: continued from previous page

(A) Example of a search over SAbDab-nano by attributes: The search retrieves all nanobody structures with data in CoV-AbDab, for which the antigen name contains the string "spike". **(B)** The first 3 entries retrieved by the query. **(C)** The structure viewer interface, which shows the annotation on the nanobody structure and download options and provides an interactive 3D visualisation of the structure.

2.5 Results

2.5.1 SAbDab and SAbDab-nano have seen fast growth in recent years

As of September 2021, SAbDab-nano contained 823 structures (see Fig. 2.3), representing 492 nanobodies with non-redundant CDR sequences. This is a more than five-fold increase over the 123 nanobody structures available when SAbDab was initially published, signifying the need for a dedicated sub-database for nanobody structures. SAbDab-nano has grown at an average 3.8 structures/week over the first 36 weeks of 2021. This is only slightly slower than the pace of growth of SAbDab (5.2 structures/week) during the year of its release (2013), further showing the growing need for a more dedicated resource. As discussed further in section 2.6, the fast growth in the number of experimentally determined nanobody structures is partially driven by the interest in nanobodies as potential SARS-CoV-2 therapeutics: 23% of nanobody structures deposited since March 2020 are structures of coronavirus-binding nanobodies.

2.5.2 The antigen types in SAbDab and SAbDab-nano differ

The majority of SAbDab-nano entries are resolved in complex with a protein antigen, 78% of entries. This is a far larger proportion than for SAbDab, where only 56% entries contain a protein antigen (see Fig. 2.3). This difference in the distribution of antigen types between SAbDab-nano and SAbDab is primarily due to a shift over last decade in the types of antibody structures which are experimentally determined; in 2013, protein antigens were present in 55% of PDB-deposited antibody structures (not counting nanobody structures), in 2020 this ratio had risen to 75%. As more than 50% of nanobody structures have been released since 2018, SAbDab-nano reflects this trend in its overall composition.

2.5.3 New database access tools enable creation of bespoke data sets and analysis of database contents

The new search features and database access tools enable the simple creation of custom data sets of antibody crystal structures for antigen specific research, as shown with the following two examples.

The creation of a data set of coronavirus binding antibodies is a topical use case. While this can now be easily achieved using the link between SAbDab and CoV-AbDab, the new keyword search feature enables the identification of the majority of these structures without the manual curation necessary for CoV-AbDab: generating a keyword query from the keywords "SARS-COV", "COV2", "COV-2" and "CORONA" retrieved 414 of the 435 SAbDab entries with annotation in CoV-AbDab as of 10/11/2021 as well as 13 additional structures, 7 of which

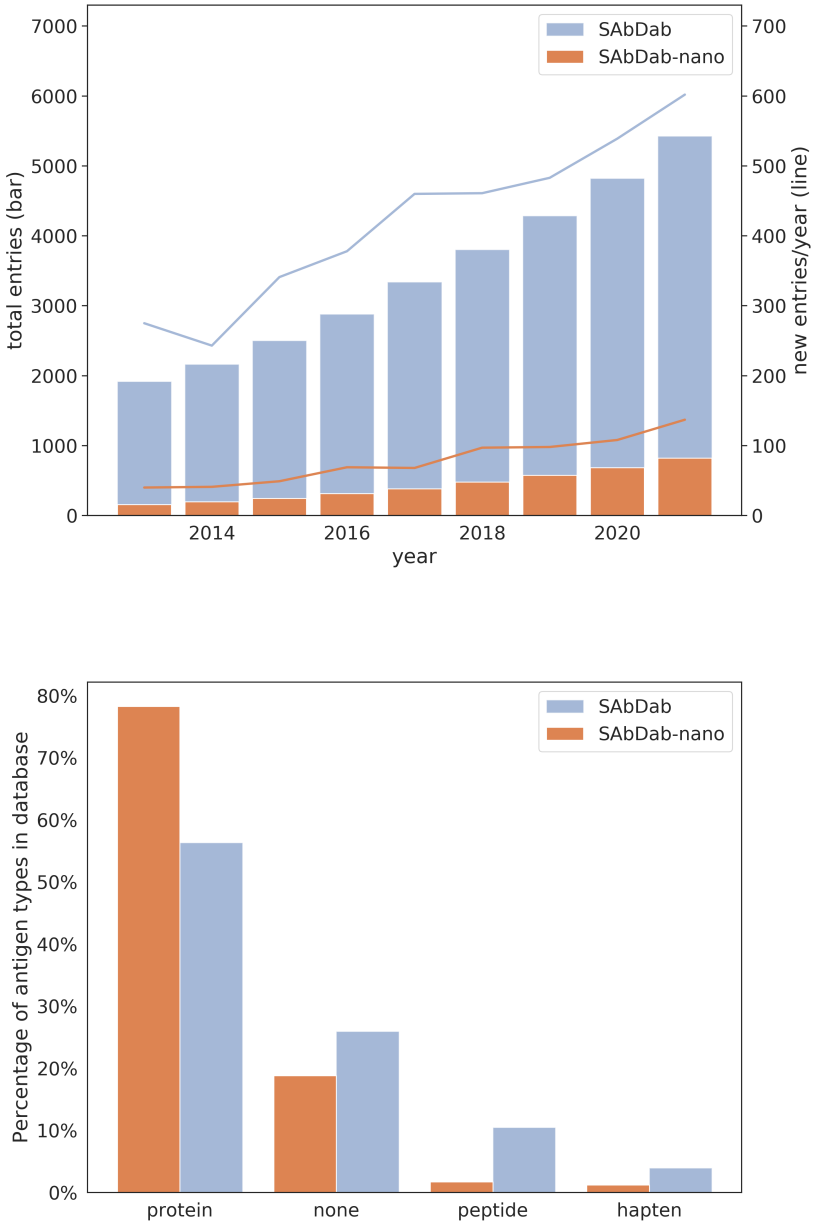

Figure 2.3: SAbDab-nano and SAbDab statistics. **(Top)** The number of entries in SAbDab and SAbDab-nano over time since the publication of the original SAbDab paper. The left y-axis and the bar plot depict the total number of entries in the respective databases at the end of the year, the right y-axis...

Caption continues on next page.

Figure 2.3: continued from previous page

...and the line plot depict the number of entries added to the databases in that year. Numbers for 2021 cover the time until 09/09/2021. Numbers for SAbDab include the content of SAbDab-nano, as SAbDab-nano is a subset of SAbDab. **(Bottom)** Antigen type composition of SAbDab and SAbDab-nano. Antigen types for which no structures exist in SAbDab-nano, but structures exist in SAbDab (carbohydrate, nucleic acid, other) are omitted. Protein here describes a polypeptide consisting of more than 30 amino acids, while peptide describes polypeptides consisting of less than 30 amino acids.

were structures of SARS-CoV-2 binding antibodies which were later added to CoV-AbDab. Therefore, the keyword search feature enables the rapid generation of antigen specific antibody structure data sets.

Similarly, a use case of interest could be the identification of all structures of therapeutic antibodies against a particular target. For example, filtering for SAbDab entries with annotation in Thera-SAbDab enables specifically retrieving the 19 structures of therapeutic antibodies binding to SARS-CoV-1 or SARS-CoV-2.

2.5.4 SAbDab-nano provides the most comprehensive database of nanobody structures

There are other databases and resources compiling nanobody data. However, to my knowledge no other resource provides nanobody structures in a continuously updated and comprehensively annotated format.

There are several databases compiling nanobody sequences (but not structures) from a variety

of data sources. INDI (Deszyński et al. 2021) contains more than 11 million nanobody sequences, including, at time of writing, sequences derived from 805 PDB structures, but does not provide experimentally resolved structures. sdAb-DB (Wilton et al. 2018) contains 1452 single-domain antibody sequences, of which 195 are derived from experimentally resolved structures, but similarly does not provide the corresponding structures.

There are two currently accessible resources containing annotated nanobody structures. Zavrtanik and Hadži (2019) compiled a static, non-redundant set of 123 nanobody structures in complex with antigens, derived from 217 PDB deposited structures. Abybank-AbDb (Ferdous and Martin 2018) contains a set of 347 nanobody structures, which can be downloaded in bulk, but over which no additional search can be performed.

Further, nanobody structures are of course present in the PDB and the IMGT/3Dstructure-DB (Ehrenmann et al. 2010) but are not annotated as such and thus cannot be retrieved trivially from these databases.

2.6 Discussion

SAbDab continues to be updated weekly and represents the most thoroughly annotated antibody structure database from which researchers can quickly create custom data sets for their studies. Following my contributions to the database, SAbDab is more powerful, with new connections to auxiliary databases that catalogue therapeutic and antigen-specific antibodies. These links will continue to be extended as more such databases become available. By implementing SAbDab-nano, an explicit nanobody-tracking sub-database, I have provided an addi-

tional resource for researchers investigating the structural properties of this emerging class of bio-therapeutics.

Through the incorporation of CoV-AbDab into SAbDab, recent trends in the deposition of antibody structures have become evident. As stated above, 7.8% (435) of all antibody structures and 8.2% (71) of all nanobody structures are of coronavirus-binding antibodies/nanobodies, the vast majority of which binds to the SARS-CoV-2 receptor binding domain. This epitope-specific bias will have to be taken into account in future studies of the antibody structural space. For example, machine learning tools trained on paratope-epitope interfaces including data deposited in the last two years in their training sets would likely learn this dataset specific bias, reducing generalisation of the resulting models. Similarly, analysis of antibody properties based on deposited structures (such as paratope topology or amino acid composition) could be skewed towards the specific properties of coronavirus-binding antibodies. Those issues can be addressed through epitope-specific data weighting strategies during model training or data analysis respectively.

There are several outstanding areas of improvement for SAbDab. Additional antigen-specific databases will improve the ability to create custom data sets in the future. Further, SAbDab currently has one entry per PDB deposited structure. However, roughly 5% of the database (259 entries) contain more than one non-redundant antibody, which is currently not immediately obvious in the SAbDab interface. Incorporating this information into the SAbDab data annotation will improve users' ability to easily search over SAbDab.

SAbDab and SAbDab-nano can be accessed freely online under a CC-BY 4.0 license at

opig.stats.ox.ac.uk/webapps/newsabdab/ or at

opig.stats.ox.ac.uk/webapps/newsabdab/nano respectively.

3 | High-throughput generation and assessment of antibody-antigen complex poses

Contents

3.1 Chapter motivation

As described in the introduction, the overall object of my DPhil research is to develop novel methods to computationally predict antibody-antigen binding, in particular exploring the use of 3D structural information alongside machine learning techniques. This chapter details the generation of predicted antibody-antigen complex structures, an analysis of the quality of these complexes and the development of DLAB-Re, a novel antibody-antigen specific docking pose rescoring tool. The work detailed in this chapter forms part of the published paper "DLAB: deep learning methods for structure-based virtual screening of antibodies" (Schneider et al. 2021b).

3.2 Introduction

In order to study the interaction between antibodies and cognate antigens on a structural level as well as to classify their capacity to form binding interactions, structures of the binding interface are required.

However, as described in section 1.3, only ca. 3200 structures of antibodies in complex with protein antigens have been experimentally determined to date (Dunbar et al. 2014) due to the cost- and time-intensive nature of experimentally determining antibody-antigen complex structures. Given the numbers of potential antibodies and antigens, there are trillions of possible antibody-antigen pairings, so it is necessary to predict (through docking, see section 1.3) complex structures that can be used to assess the binding potential of a given antibody-antigen

pairing.

To conduct such docking studies, structures of antibodies and target antigens need to be generated. Antibody structures can either be generated experimentally (as detailed in chapter 2, there are over 5500 experimentally determined antibody structures accessible in SAbDab as of November 2021) or computationally, with several state-of-the-art antibody modelling tools being available, e.g. ABodyBuilder (Leem et al. 2016), Rosetta (Weitzner et al. 2017) and Kotai Antibody Builder (Yamashita et al. 2014). For an overview of antibody modelling, see section 1.2.2.

In a high-throughput screening setting, where thousands of antibody structures are screened to find binders against a specific target antigen, antibody structures need to be modelled computationally. Antibody modelling is much faster than experimentally determining structures (ABodyBuilder for example can generate an antibody model in 30 seconds) and therefore enables the exploration of a larger area of antibody sequence space.

Target antigen structures can also be generated experimentally or computationally. For high-throughput screening of antibodies, the use of experimentally determined structures of antigens is viable since only one accurate antigen structure would be required but accurate models can also be used. Recent advances in computational structure prediction (e.g. Jumper et al. (2021), see also section 1.2.2.1) suggest that the highly accurate computational prediction of target antigen structures will soon be routinely available.

There are two main approaches to computationally generate antibody-antigen complexes: Fast rigid body docking and slower, but potentially more accurate flexible docking. In rigid body

docking (e.g. ZDOCK (Pierce et al. 2011)), the input structures of interaction partners remain unchanged, with the docking algorithm outputting translation and rotation matrices to generate docking poses. In flexible docking approaches (e.g. SnugDock (Weitzner et al. 2017)), the relative coordinates of the sidechain and/or backbone atoms of each interaction partner are updated as part of the docking process (Porter et al. 2019). For more detail on different docking approaches, see section 1.3.

Both approaches generate a list of poses for each input pair of binding partners. These are then ranked to identify those poses amongst all generated poses which are considered near-native by the docking algorithm. However, this ranking often fails to promote acceptable poses to the top of the pose list, as demonstrated throughout the Critical Assessment of PRedicted Interactions (CAPRI) competitions held over the last decade (Desta et al. 2020).

To improve the ability of docking programs to identify high-quality poses, rescoring methods assess the generated poses with scoring functions different from the internal scoring potentials in the docking algorithm. These rescoring approaches can be domain specific (e.g. Krawczyk et al. (2013)), giving an advantage over general scoring functions employed by most docking programs. Recently, machine learning based docking pose scoring models have been developed by several groups (e.g. Eismann et al. (2020a) and Wang et al. (2019b), see also section 1.6.2.3). In this chapter I describe the DLAB-Re tool, an antibody-antigen domain-specific machine learning docking pose scoring model. The aim of DLAB-Re is to identify high-quality poses from a list of docked poses of antibody and antigen complexes by learning the features of well-docked poses from a diverse dataset of docked antibody-antigen complexes. DLAB-Re is the

first machine learning based rescoring tool specifically trained on the antibody-antigen domain as well as the first such tool to be trained explicitly to assess docked poses of computationally generated antibody structures.

3.3 Methods

3.3.1 Data set of antibody-antigen complex structures

As described in chapter 2, the structural antibody database (SAbDab) (Dunbar et al. 2014) contains an up-to-date collection of all experimentally determined antibody structures deposited in the PDB (Berman et al. 2000). Training sets of antibody and antigen structures for the experiments detailed in this chapter, as well as those detailed in the following chapters, were derived from a snapshot of SAbDab taken on 19/12/2018. Using the data annotation deposited alongside the structure, I selected a data set of structures of VH-VL paired antibodies in complex with protein or peptide antigens with a resolution better than 3.0 Å. This cutoff represented a reasonable balance between data quality and dataset size, with similar resolution cutoffs used in several recent studies compiling protein structure data sets for machine learning (e.g. Townshend et al. (2019) and Jumper et al. (2021)). The data set was not filtered by species, since antibody structure is highly conserved across species (Barré et al. 1994). The data set consisted of 1216 structures of antibody-antigen complexes, of which 759 were non-redundant. Redundancy was defined via the sequence of the antibody CDRs. The antibodies were IMGT numbered (Lefranc et al. 2003) using ANARCI (Dunbar and Deane 2015) and the CDR residues were identified as those at positions 27-38, 56-65 and 105-117 on the heavy and light chain

respectively. Antibodies were considered non-redundant for the data set creation if their concatenated CDR sequences were distinct. The PDB accession codes for this data set, which I will refer to as the crystal structure data set in the following, are detailed in Appendix Table A.2.

3.3.2 Generation of antibody models

Using the SAbDab Python API, I extracted the amino acid sequences of the antibodies in the crystal structure data set. Models of the antibodies were generated using ABodyBuilder in parallel execution mode. ABodyBuilder was set up not to include any templates that are fully identical to the input sequence so that perfect models could not be built. In its default setting, PEARS (Leem et al. 2018), the sidechain modelling component of ABodyBuilder, rejects side chains for which no conformation without clashes to the rest of the input structure can be found and does not include those side chains in the final model. However, binding is driven mainly by sidechain interactions. I therefore extended the PEARS source code to model sidechains with the least clashing confirmation if only sterically clashing confirmation were found. This force-fit version of PEARS was used throughout the research detailed in this thesis. The generated antibody models were numbered using the IMGT numbering scheme for model quality assessment and the Chothia or IMGT scheme for downstream processing. This set of ABodyBuilder model antibodies is referred to in the following as the model data set.

3.3.3 Assessment of model quality

Antibody model quality was assessed via calculation of the RMSD of the backbone atoms (N, Cα and the atoms in the backbone carboxyl group) of the model to the corresponding antibody in the PDB-deposited structure. This metric was chosen in order to enable the comparison of the model quality to previous studies using the same metric (Leem et al. 2016). For this purpose, I superimposed the antibody model and crystal structure by their framework regions using the SVDSuperimposer class in the Biopython package (Cock et al. 2009). The RMSD was then calculated over the entire Fv region, across all CDRs or for specific CDRs using this framework-based superimposition.

3.3.4 Generation of antibody-antigen complex poses

3.3.4.1 Generating docked poses

For the reasons outlined in section 1.3.3, I used ZDOCK 3.0.2 (Pierce et al. 2011) to generate 2000 docking poses for each antibody in both the crystal structure data set and model data set against their cognate antigen. For both cases, the antigen in the experimentally determined structure was used. Prior to docking, I removed crystal waters from the antigen to avoid biasing the docking algorithm too strongly towards the native pose. While for few cases, this might have removed water molecules involved in the binding interaction, in a realistic high-throughput docking scenario using modelled antibody structures, this information would not be available.

3.3.4.2 Determining paratopes and epitopes for information driven docking

Antibody-antigen docking can benefit from the inclusion of information about the paratope and epitope into the docking algorithm (Ambrosetti et al. 2019). To guide the docking process, I determined residues belonging to the antigen epitope or the antibody paratope and marked residues not belonging to either so as to exclude them from the interaction site during the docking pose generation.

The antigen epitope was defined by separating the antibody and antigen in the crystal structure and calculating surface exposed residues for each binding partner separately using the PSA algorithm (Lee and Richards 1971). Residues with relative surface accessibility over 7.5% according to PSA were considered to be surface residues, using the same cut-off as Raybould et al. (2019b). All atoms belonging to surface-exposed residues on the antigen less than 4 Å from a surface-exposed residue on the antibody were considered part of the epitope for the purpose of docking. To generate the allowed docking interface, all atoms belonging to a surface-exposed residue on the antigen within 4 Å of the defined epitope were included to model a realistic level of access to the epitope. The paratope of crystal structure antibodies was defined in the same way, using the interacting residues from the crystal structure.

For modelled antibody structures, two paratope prediction strategies were tested. The first strategy relied solely on antibody sequence numbering: the paratope was defined using a CDR definition (either IMGT or Chothia) and marking the CDRs per that definition and zero, two or four residues to either side of each CDR as the paratope (these strategies are in the following referred to as IMGT/Chothia_0/2/4 respectively). In the second strategy, the paratope was pre-

dicted using Parapred (Liberis et al. 2018), a state-of-the-art paratope prediction tool based on a recurrent neural network, using two scoring strategies: Either all residues with a Parapred score over 1% were considered paratope residues (representing a permissive paratope definition, only excluding residues predicted to be highly unlikely to be involved in the binding interface) or all residues with a Parapred score over 67% (the score cut-off suggested in the Parapred paper, representing a more conservative paratope definition) and all neighbouring residues (less than 4 Å heavy atom distance) were considered paratope residues (these strategies are in the following referred to as Parapred_1/67 respectively).

3.3.5 Docking quality assessment

The quality of individual docked poses was primarily assessed through the calculation of the fraction of native contacts (the residue-residue contacts between antibody and antigen present in the experimentally determined complex structure) retained by the docked pose (*fnat*) as defined for the assessment of the CAPRI competition (Wodak and Méndez 2004). To calculate this metric, I identified pairs of residues for which any resolved atom of the residue was located within 5 Å of an atom in the binding partner, considering these residue pairs native contacts. The *fnat* score is then calculated as:

$$f_{nat} = \frac{|C_p \cap C_n|}{|C_n|} \tag{3.1}$$

$$C_d = \{c_{d,1}, ..., c_{d,i}\} \tag{3.2}$$

$$C_n = \{c_{n,1}, ..., c_{n,i}\} \tag{3.3}$$

where $c_{d,i}$ is the i-th contact in the docked pose and $c_{n,i}$ is the i-th contact in the native, experimentally determined complex structure.

Docking pose quality assessment was further conducted using the interface RMSD (iRMSD) and ligand RMSD (lRMSD) metrics according to the CAPRI definitions. iRMSD is the RMSD across all backbone atoms belonging to residues within 10 Å of the binding partner, calculated after aligning the crystal structure and the docked pose by those residues. lRMSD is the backbone RMSD of the binding partner designated as the ligand after aligning the crystal structure and the docked pose by the binding partner designated as the receptor. The metrics are further detailed in section 1.3.3.1 and examples of well and poorly docked poses are provided in Fig. 1.9.

3.3.6 Knowledge-based rescoring of docked poses

Knowledge-based docking pose rescoring was carried out using a statistical scoring potential for antibody docking poses described by Krawczyk et al. (2013). The potential is derived by calculating a precision score $P(a \rightarrow b)$ for each possible pair of contacting amino acids *a* on the antibody and *b* on the antigen as per the following equation:

$$P(a \rightarrow b) = \frac{TP(a \rightarrow b)}{TP(a \rightarrow b) + FP(a \rightarrow b)} \tag{3.4}$$

where $TP(a \rightarrow b)$ is the number of times the contact between amino acids *a* and *b* is seen in the top 100 ZDOCK created poses for all docking runs in the model data set and is also present in the experimentally determined crystal structure (the true positives) and $FP(a \rightarrow b)$ is the

number of times the contact is seen in the ZDOCK created poses but not in the experimentally determined crystal structure (the false positives). The ZDOCK output was re-ranked by the sum of the precision scores for all interacting amino acid pairs in each pose (as defined above). In the following, I refer to this approach as "IPatch reranking" in reference to the paper in which it was introduced (Krawczyk et al. 2013).

3.3.7 Creating dataset splits

The binding specificity and correspondingly the binding mode of antibodies is defined mainly by the CDR sequences. To avoid overlap between the binding modes of antibodies assigned to the train and test sets, CD-HIT (Li and Godzik 2006) was used to cluster antibodies by their identity across the IMGT defined CDR sequences, using a clustering cut-off of 90% sequence identity with the default CD-HIT settings. For training of DLAB-Re, a five-fold clustered cross-validation split was used, dividing the clusters into five folds with equal cluster numbers and assigning all members of one cluster to the same train or test set.

3.3.8 Discretisation of structure data

Following the gridding method first published by Ragoza et al. (2017) and released as the `libmolgrid` package by Sunseri and Koes (2019), the docking poses were prepared for input into neural networks by discretising the atom information into four-dimensional grids, where three dimensions describe the spatial arrangement of the interaction site and the fourth dimension is used to indicate atom types (see Fig. 3.1 and Appendix A.1).

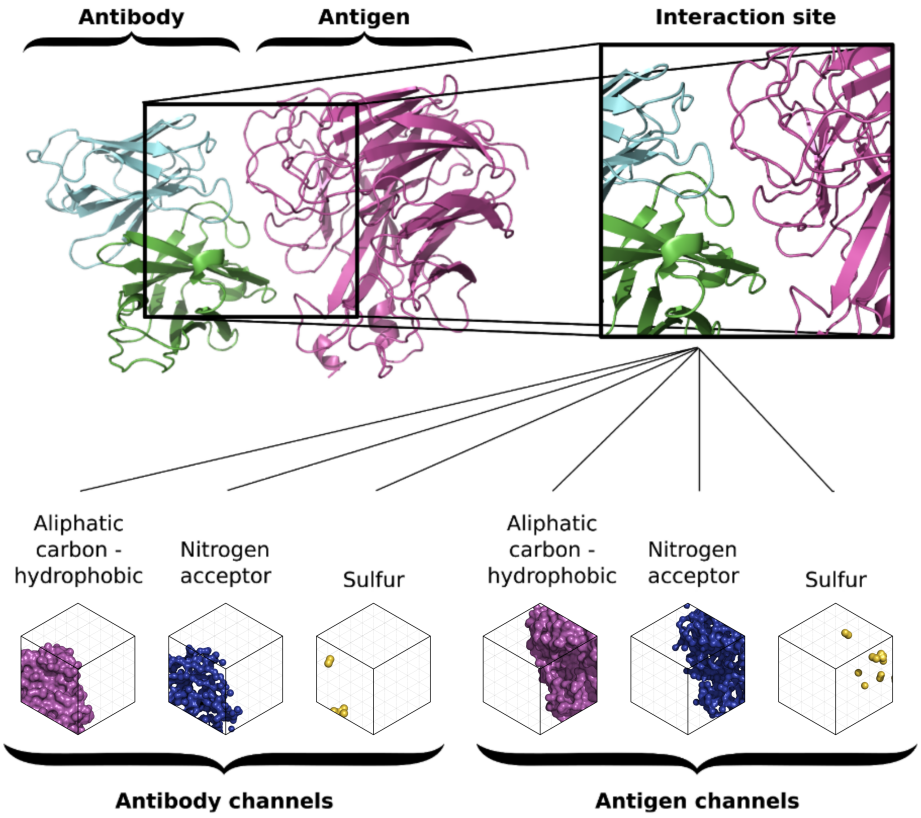

Figure 3.1: The binding interface discretisation algorithm. (**Top**) The interface is defined as a 48 Å cube centered on the interaction center between antibody and antigen. (**Bottom**) The atoms occupying the interface in the docking pose are discretised using the `libmolgrid` Python API into 3D arrays of atom type densities (14 types for both antibody and antigen, of which three are depicted here as an example). The full list of types can be found in Appendix A.1.

The centre of the interaction site of docked poses was calculated using the PSA algorithm to define surface residues as detailed in section 3.3.4.2 by averaging the coordinates of all surface-exposed atoms within 4 Å of a surface-exposed atom on the interaction partner on both the antibody and the antigen and taking the mean of the two center points. Surface exposed atoms were determined for each interaction partner in isolation. Poses which after docking had no interactions under 4 Å were discarded.

During model optimisation, different grid dimensions (24 Å, 40 Å and 48 Å) were tested as well as different resolutions of the input grid (1 Å and 0.5 Å).

For the final model, the grid was set to contain all atoms that were within 24 Å of the interaction centre. These interaction site specifications were found to cover on average 96% of all interacting atoms on both antibody and antigen (see Fig. 3.2). The grid resolution was set to 0.5 Å, leading to a total grid size of 96^3 voxels.

3.3.9 DLAB-Re training

3.3.9.1 Task setup

Each antibody in the crystal structure and model data set was docked against its cognate antigen as described in section 3.3.4. For each antibody-antigen pairing, the 500 highest ranked poses generated by ZDOCK were annotated with a single, continuous quality score, their *fnat* score. The top 500 poses were chosen since they provide sufficient coverage of the *fnat* space for individual docking runs and importantly include high *fnat* poses in docking runs, as demonstrated by comparing cumulative distributions of *fnat* values using different numbers of poses across

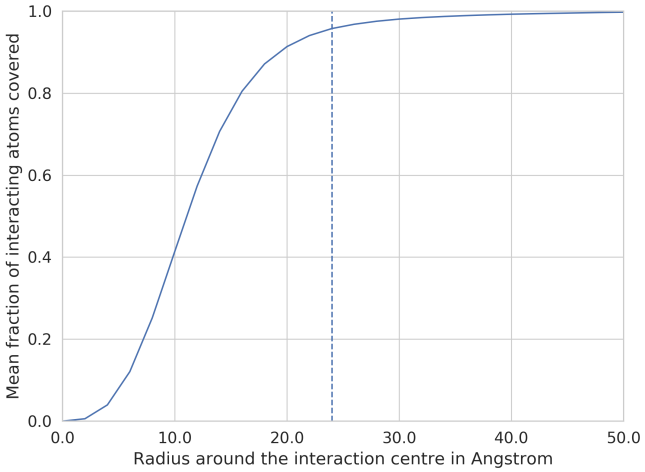

Figure 3.2: The fraction of interacting atoms included in the interaction box depending on the radius around the interaction center. The dashed vertical line indicates 24 Å, the radius used throughout the majority of this thesis.

four docking runs (see Fig. 3.3).

Models were trained to recapitulate the *fnat* score given a docking pose and the top 500 poses were re-ranked according to the predicted *fnat* score. The performance of the models was evaluated by assessing the enrichment of poses with *fnat* $>$ 0.3, 0.5 and 0.7 in the top 1, 5, 10, 20, 30, 40 and 50 poses respectively. The first two *fnat* values correspond to the CAPRI medium (0.3) and high (0.5) quality categories, I added the 0.7 threshold to assess high-quality poses with more granularity.

3.3.9.2 Model architectures

Several model architectures were tested during the development of the rescoring model DLAB-Re.

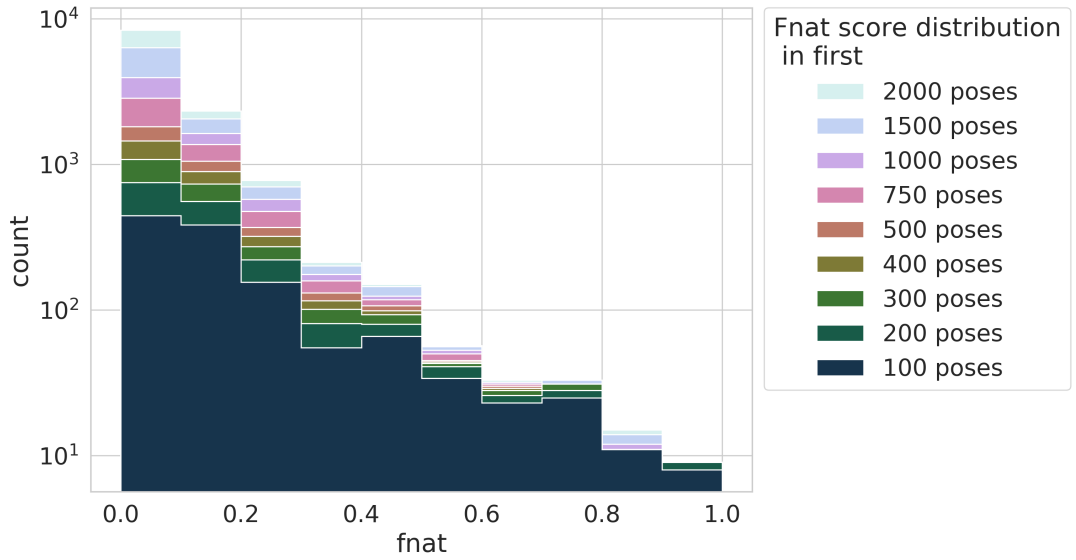

Figure 3.3: *fnat* values encountered across the ZDOCK ranked poses for four antibodies (PDB IDs 1a14, 1adq, 1afv, 1ahw) in the crystal structure data set. The plot shows the distribution of *fnat* scores over the top 100 to 2000 poses generated and ranked by ZDOCK, indicating the distribution of *fnat* values of poses added by considering increasing amounts of poses as ranked by ZDOCK. Adding poses beyond the first 500 poses primarily contributes poses with low *fnat* values and the first 500 poses yield sufficient coverage of the range of *fnat* values.

The basic architecture employed for DLAB-Re consisted of a 3D Convolutional Neural Network (CNN) composed of three consecutive convolutional layers and BatchNorm layers, followed by a flattening layer and a fully connected layer (see Fig. 3.4). This was inspired by the architecture used successfully by Ragoza et al. (2017) on small molecule docking scoring.

The architecture detailed above was implemented with two output layer options: firstly, a linear output layer was used to attempt to directly regress the pose *fnat* score. Secondly, a softmax layer was used to generate a probability distribution over 11 *fnat* intervals (ten steps of size 0.1 over [0, 1] and one bin for *fnat* 0.0 poses), which were then used to generate a *fnat* prediction by averaging the probabilities predicted for each interval multiplied by the upper bound of the interval. The former of the two options failed to converge during training so for the rest of this thesis, I use the latter option. While I did not further investigate the lack of convergence of the regression approach, it could have been caused by insufficient exploration of training and model hyperparameters.

3.3.9.3 Model training

For any given antibody-antigen pairing, there are considerably more poses with low *fnat* scores (poorly docked poses) in the top 500 ZDOCK poses than high *fnat* poses. To avoid biasing the network towards assigning low *fnat* values to all poses, I used a stratified sampling scheme, sampling poses from each interval at the same rate during training (but not during testing).

During training, the input data was augmented by random rotation around the interaction centre, followed by random translations along the x, y and z axes between $-2\,\text{Å}$ and $2\,\text{Å}$. This

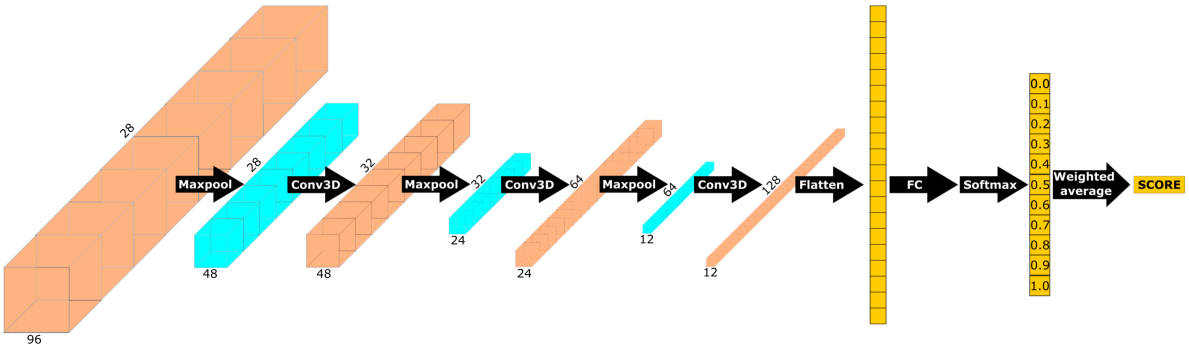

Figure 3.4: DLAB-Re model architecture. The input grid (depicted at 48 Å side length and 0.5 Å resolution) is passed through three convolutional layers, each consisting of a max-pooling operation with stride and kernel size 2 (output from this operation is depicted in blue) and a 3D convolution with kernel size 3, followed by ReLu activation and BatchNorm normalisation. The output of the last convolutional layer is flattened and passed through a fully connected layer with output size 11, corresponding to the 11 bins into which the *fnat* of docking poses is classed.

augmentation procedure, which was introduced by Ragoza et al. (2017), was essential to avoiding model overfitting. Models were trained to convergence for 200,000 parameter update steps using categorical cross-entropy loss (for the multi-class classification formulation of the docking rescoring task in combination with the softmax output layer described above) or smooth L1 loss (for the regression formulation of the task in combination with the linear output layer) and the rectified ADAM optimiser, which was used in order to increase convergence speed and stability in early training compared to the native ADAM optimiser.

3.3.10 DOVE rescoring

I compared DLAB-Re to the DOVE method for CNN-based docking pose ranking (Wang et al. 2019b). DOVE uses a similar CNN architecture to DLAB-Re, using four atom types alongside

contact potentials in a grid of side length 20 Å or 40 Å and resolution 1 Å to generate an output probability, predicting how likely an input pose is to fall into the CAPRI category acceptable or better. Input file preparation and score generation were performed according to the tutorials on the author's GitHub page (Wang et al. 2021b). As detailed on the author's GitHub page, only the GOAP and ATOM20/ATOM40 scores were used, as the IT-scores were unavailable.

3.3.11 DLAB-Re thresholding

To identify antibody-antigen pairings where all 500 docking poses were of low quality, I determined the highest DLAB-Re score given to any of the top 500 poses generated by ZDOCK for each antibody-antigen pairing. This score (DLAB-Re-max) was used to discard particular pairings by ranking all pairings by their DLAB-Re-max score and discarding the bottom 40%, 60% or 80%. To contrast the performance of ZDOCK, DOVE and IPatch on the same task, this score thresholding was also applied using the ZDOCK output score, IPatch score and the DOVE ATOM40+GOAP score.

3.3.12 Statistical testing

Throughout this chapter, for statistical significance testing of the difference between means of the best *fnat* in top ten ranked poses, I used the two-tailed t-test implementation in the scipy Python package (Virtanen et al. 2020). For statistical significance testing of the ranking performance of DLAB-Re, I considered the ratio of antibody-antigen pairs for which a pose with a specific *fnat* is found in the top ten poses a Poisson rate and calculated p-values using the implementation of the test described in Gu et al. (2008) in the statsmodels Python package

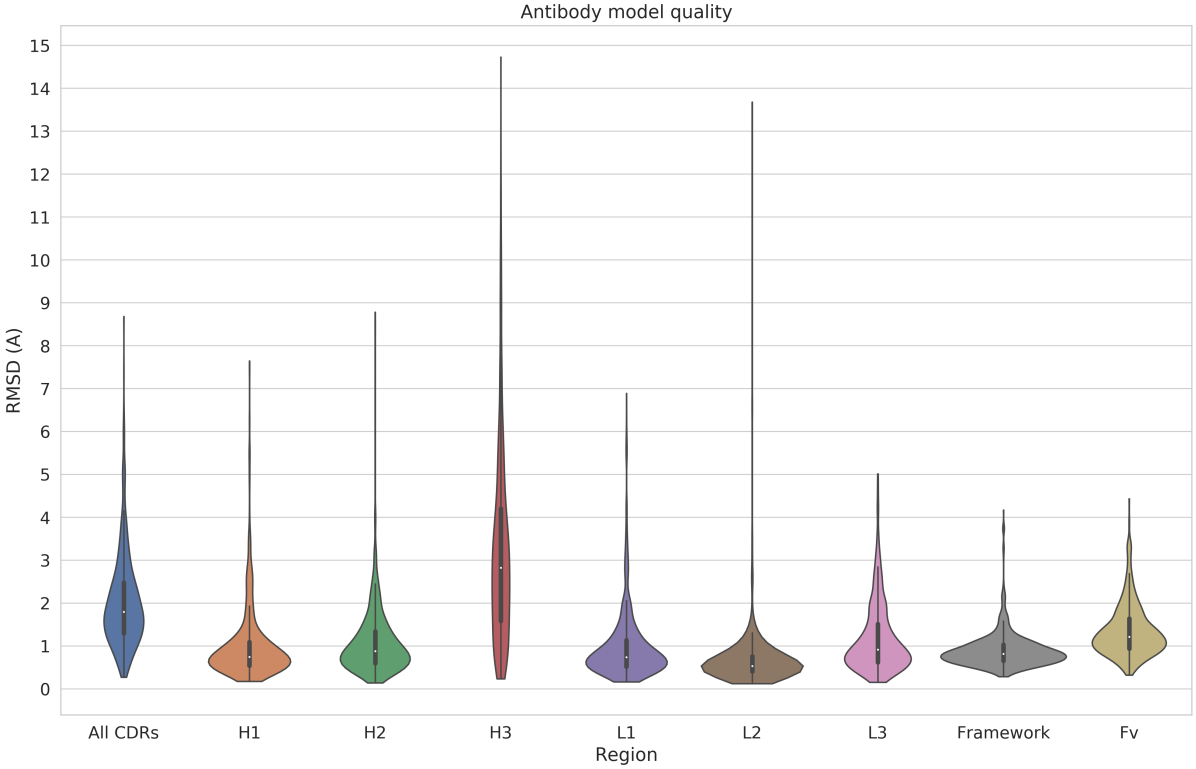

Figure 3.5: Antibody model quality assessment by backbone RMSD. The RMSD values were calculated as specified in section 3.3.3. As expected, the framework and the non-H3 CDR regions can largely be modelled with sub-Angstrom RMSD, while the more variable CDR H3 can pose difficulties during modelling, driving the higher spread in RMSD across all CDRs compared to the non-H3 CDRs. For reference, the regions are visualised in Fig. 1.2.

(Seabold and Perktold 2010). Correspondingly, I approximated the standard deviation of the ratio as $\frac{\sqrt{c}}{n}$, where c is the count of antibody-antigen pairings for which a pose with a specific *fnat* is found in the top X poses and n is the total number of pairings assessed.

3.4 Results

3.4.1 ABodyBuilder performance assessment

To confirm that the antibody models in the model dataset are of sufficient quality for downstream analysis, I calculated the backbone RMSD of all modelled antibodies to the corresponding experimentally determined co-crystal structures (see Fig. 3.5). The models had a mean RMSD of 0.9 Å across the framework region and 1.3 Å across the full Fv, slightly outperforming a previously conducted benchmark of the ABodyBuilder software (Leem et al. 2016). This difference is probably due to the larger databases of experimentally determined structures that the software is able to draw from for template selection (2770 in 2016 when ABodyBuilder was published vs. 4110 in 2019 when this analysis was performed). As in that benchmark, the CDR H3 loop proved difficult to model accurately due to its high structural and sequence variability. The modelling accuracy achieved here however is in line with comparable antibody modelling methods (Marks et al. (2017), see also section 1.2.2).

3.4.2 Redocking crystal structures yields highly accurate docked poses

In order to establish a baseline for ZDOCK performance on antibody and antigen crystal structures, I re-docked the complexes in the crystal structure data set and quantified the docking performance through the *fnat* score of the docked poses (see Fig. 3.6). ZDOCK yielded high-quality docked poses, ranking at least one pose with *fnat* > 0.5 in the top ten poses for 93% of the pairings.

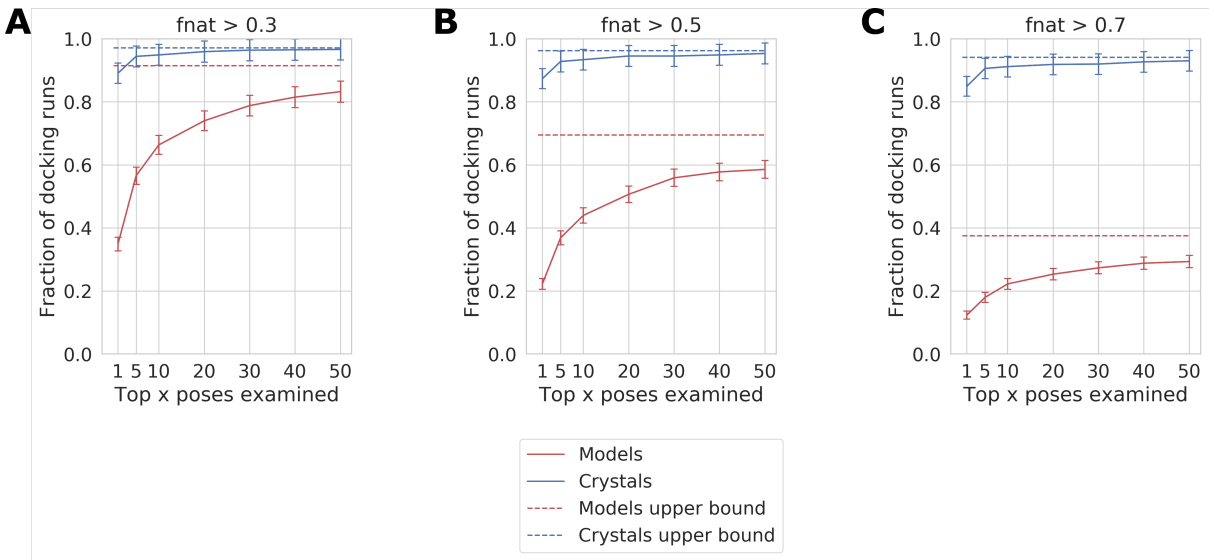

Figure 3.6: Comparison of ZDOCK runs on crystal structures and models of cognate antibody-antigen pairs. Each of the three plots shows the fraction of antibody-antigen pairings which have at least one pose meeting the threshold *fnat* score 0.3 (A), 0.5 (B) or 0.7 (C) in their x highest ranked poses. The dashed line indicates the fraction of docks which have such a pose in their top500 poses (the upper bound of the pose ranking algorithm).

3.4.3 Model docking tends to yield lower-quality poses

Using the crystal structure re-docking approach detailed above, I next investigated the ability of ZDOCK to generate accurate complex poses using ABodyBuilder generated models and the crystal structure of their cognate antigens. This approach yields on average lower quality docked poses. Here, ZDOCK created a pose with *fnat* > 0.5 in the top ten poses for only 44% of antibody-antigen pairings, but 70% of pairings had a pose with *fnat* > 0.5 in their 500 highest ranked poses (see Fig. 3.6), indicating that while docking modelled antibodies with ZDOCK is less effective at promoting near native poses to the top of the ranking, it still generates such poses, highlighting the potential for a domain-specific rescoring model.

I investigated the influence of different paratope definitions for docking purposes (defining the residues on the antibody which can be included in the docking interface). For this purpose, I compared two antibody numbering schemes and CDR definitions (IMGT and Chothia) as described in section 3.3.4.2. The two definitions yielded comparable docking results, with the best results achieved by using definition IMGT_2, setting the allowed docking interface to the IMGT defined CDR residues and two residues on either side of the CDRs (see Fig. 3.7). While Chothia_2/4 and IMGT_2/4 performed similarly, IMGT_0 performed worse than the other tested paratope definitions. This was likely due to the penalty applied by ZDOCK to docked complex interfaces including residues marked as non-interacting, highlighting the importance of the inclusion of surrounding residues in the paratope definition for docking purposes, as used by the other paratope definitions.

To investigate the performance of an established paratope prediction tool on the same task, I

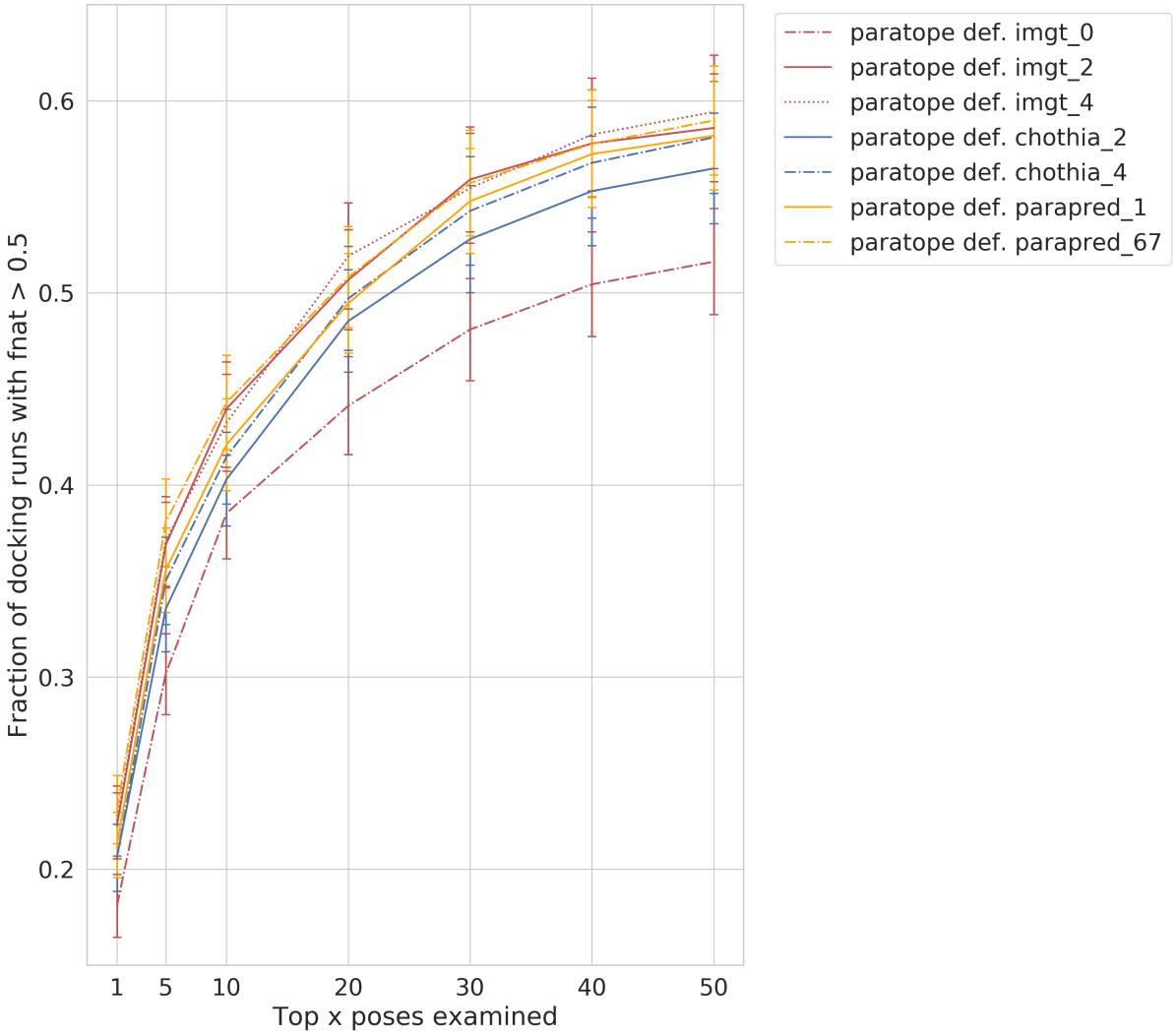

Figure 3.7: Comparison of several paratope docking contact area definitions. Definitions Parapred_1 and Parapred_67 (orange lines) use the Parapred output at 1% or 67% score threshold to determine the paratope for the purpose of defining the allowed interaction area during docking as described in section 3.3.4.2. The remaining definitions use the Chothia (blue lines) or IMGT (red lines) CDR definition, including the CDRs and zero, two or four residues to either side of each CDR in the allowed interaction site. The approaches are compared using the fraction of antibody-antigen pairings for which the top x poses as ranked by ZDOCK contain at least one pose with *fnat* over 0.5.

used the recurrent neural network based paratope predictor Parapred to define the paratope as described in section 3.3.4.2. Both a Parapred score of 1% without adding neighbouring residues to the docking contact area as well as using the 67% cut-off and adding neighbouring residues to the allowed docking contact area yielded comparable docking results to using the numbering scheme definitions (see Fig. 3.7), with no statistically significant difference between the strategies IMGT_2, Parapred_1 and Parapred_67. Since using a numbering scheme method is easier to implement in a high-throughput pipeline as well as being less prone to unexpected prediction behaviour, I decided to use a numbering scheme method for the following experiments. Since the IMGT numbering scheme is further easier to transfer to other immune molecule types, the results quoted above as well as all further antibody model docking results have been generated using the IMGT_2 strategy.

I further investigated the impact of model quality in the interface region on the performance of ZDOCK. For this purpose, I performed the analysis as above, stratifying the antibody models by their RMSD across all six CDR regions as described in section 3.3.3. I observed a strong relationship between docking performance and interface model quality. In the overall dataset, 44% of all model antibody-antigen pairings had a pose with $fnat > 0.5$ in their ten highest ranked poses. This increased to 73% of pairings when considering only cases where the antibody model had a CDR RMSD of 1.0 Å or lower (see Fig. 3.8). This result indicated that high-throughout antibody-antigen docking using computationally modelled antibody structures could be improved using recently published, fast machine learning based antibody modelling tools (e.g. Abanades et al. (2021), see also section 3.5).

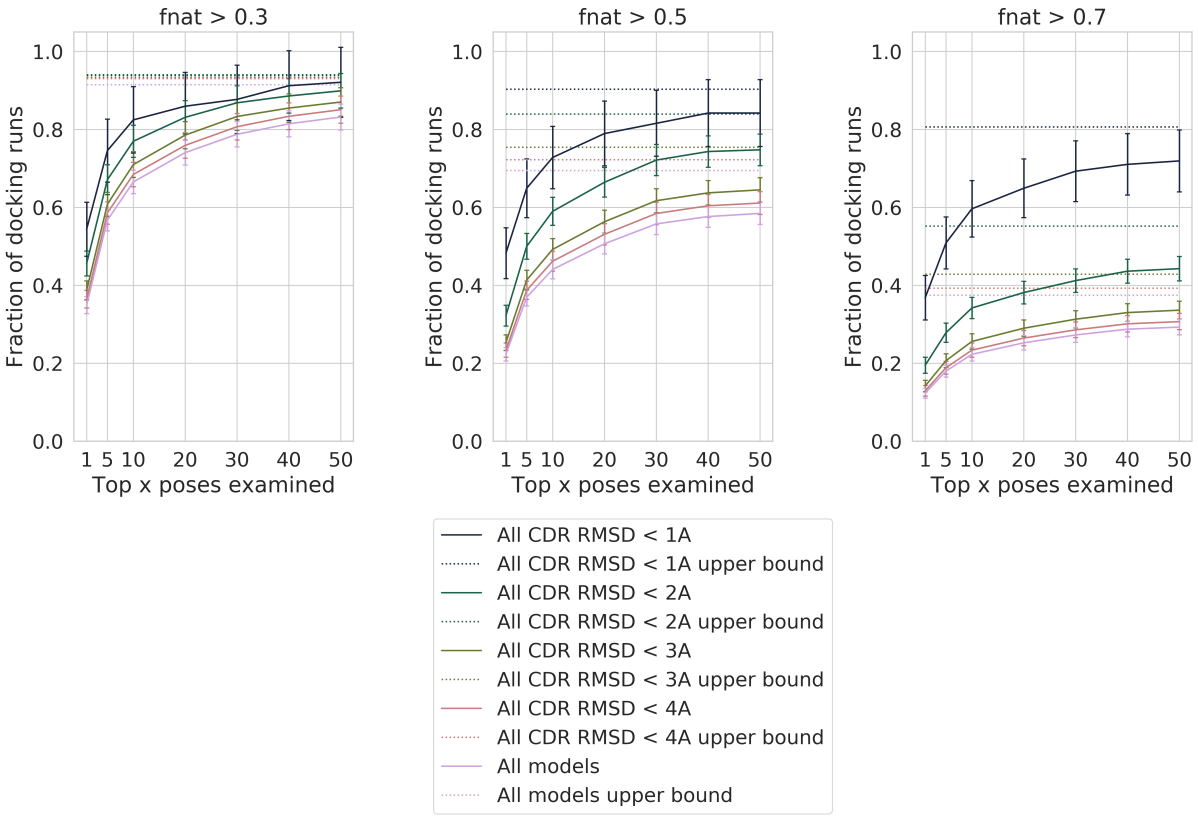

Figure 3.8: The relationship between antibody model quality and ZDOCK performance. The cognate antibody-antigen pairings are stratified by the antibody model CDR RMSD and the fraction of antibody-antigen pairings for which the top x poses as ranked by ZDOCK contain at least one pose with *fnat* over 0.3 (A), 0.5 (B) or 0.7 (C) respectively (solid lines) as well as the fraction of pairings for which a pose meeting the target *fnat* is present in the top 500 poses (dashed line). Lower CDR RMSD, indicating higher model quality in the binding interface, is linked to better docking performance.

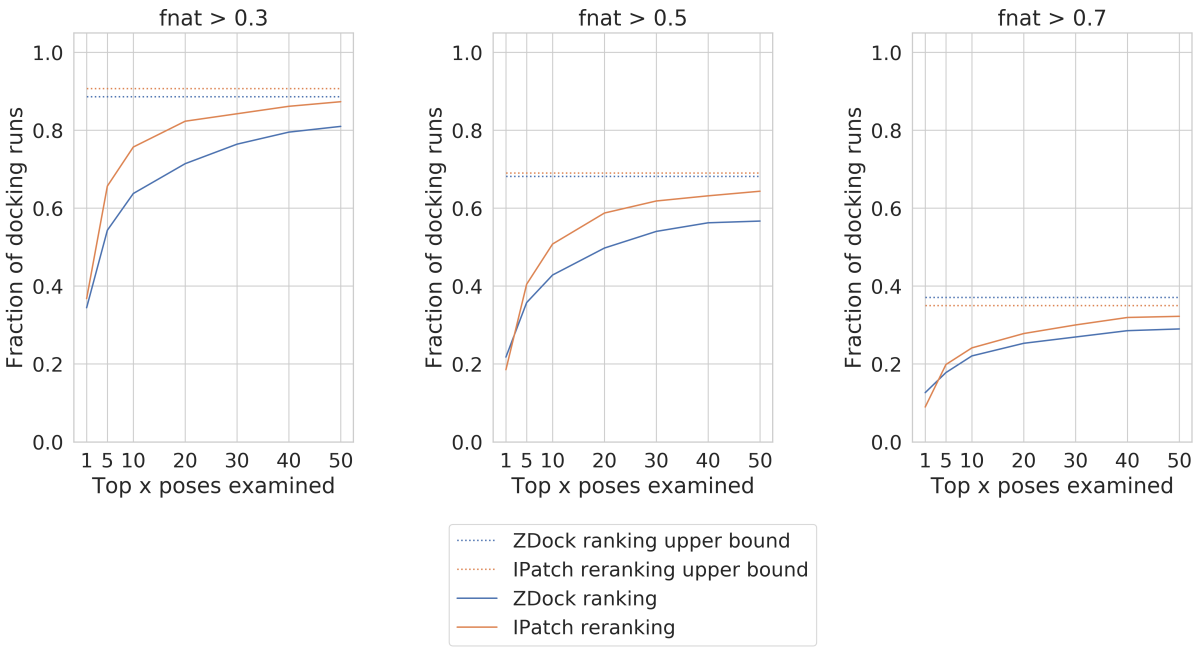

Figure 3.9: Performance of IPatch reranking on ZDOCK docking poses. The reranking scheme improves the ZDOCK ranking significantly, increasing the percentage of antibody pairings for which at least one pose with *fnat* > 0.3 (A), 0.5 (B) and 0.7 (C) is in the top ten poses by 14%, 15.5% and 8.5% respectively.

3.4.4 A knowledge-based rescoring potential can improve ZDOCK pose ranking

As ZDOCK ranks antibody model poses poorly, but generally produces high-quality poses in the top 500 poses, I initially aimed to implement a method able to improve the ZDOCK ranking performance. For this purpose, I re-implemented the statistical re-scoring potential method introduced by Krawczyk et al. (2013) and described in section 3.3.6 (IPatch reranking).

Using this approach, the ZDOCK ranking can be improved (see Fig. 3.9). IPatch reranking increases by 15.5% the percentage of antibody-antigen pairings with a pose with *fnat* > 0.5 in

the top ten poses.

3.4.5 DLAB-Re can improve ZDOCK docked pose ranking

Following on from the above results, I investigated the ability of a deep learning approach to achieve effective docking pose ranking. DLAB-Re is a CNN trained to predict the *fnat* of docked poses of antibody-antigen pairings. In order to determine the ability of DLAB-Re to improve docking pose ranking, I ranked the top 500 docked poses generated by ZDOCK for each antibody-antigen pair by the predicted *fnat* value, using the clustered, cross-validated train-test procedure set out in section 3.3.7.

This rescoring procedure improves upon the performance of ZDOCK ranking to a similar extent to the IPatch reranking tested before. On the crystal structure data set, DLAB-Re recapitulates the ranking performance of native ZDOCK. On the model data set, DLAB-Re increased the percentage of antibody-antigen pairings for which a pose with *fnat* > 0.5 is ranked in the top ten poses by 16% (p=0.047) (see Fig. 3.10 C), significantly increasing the mean best *fnat* in the top ten poses from 0.46 to 0.50 (p=0.002). In Figure 3.11, I show two antibody-antigen pairings for which DLAB-Re strongly increases the *fnat* of the best pose in the top ten ranked poses.

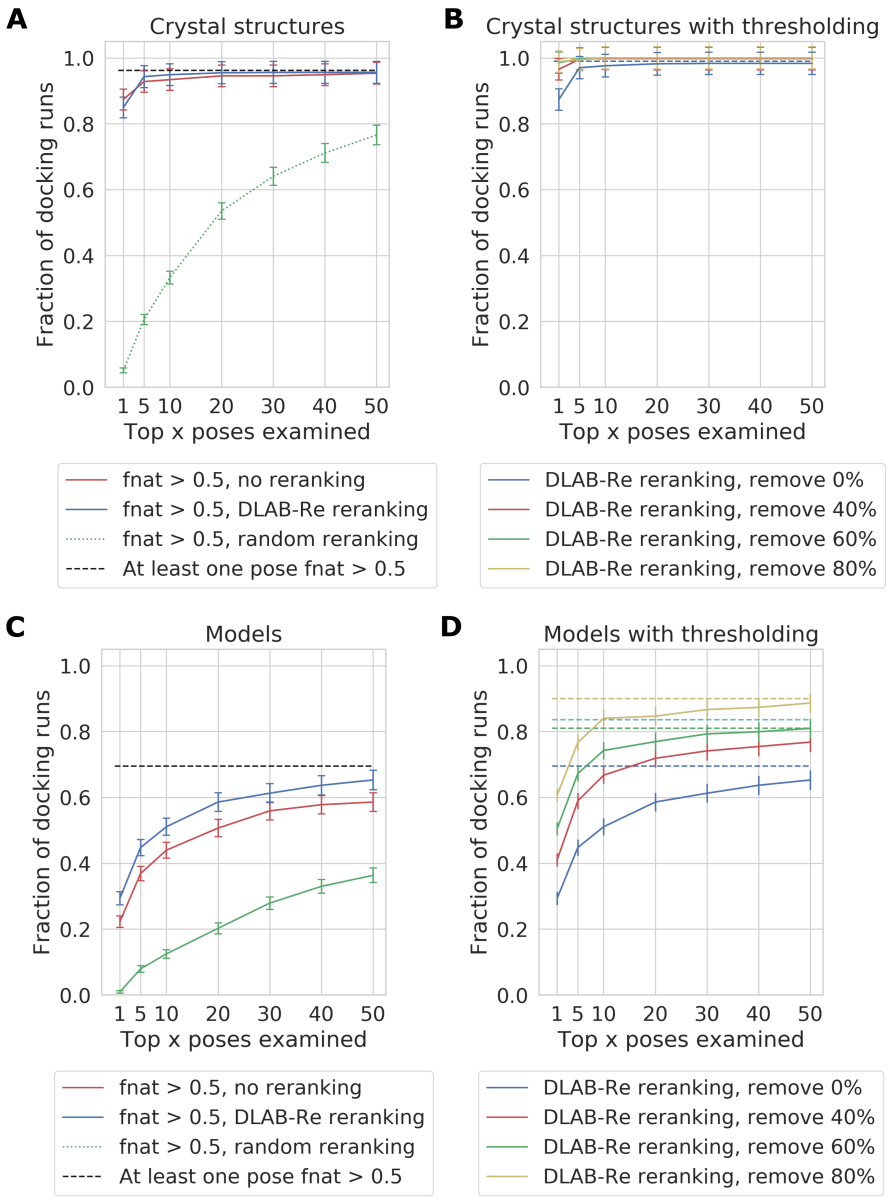

Figure 3.10: DLAB-Re improves docking performance on the crystal structure data set (**A, B**) and the model data set (**C, D**). On crystal structure data, ZDOCK ranking ("no reranking") and DLAB-Re reranking perform similarly and well, with DLAB-Re slightly but not significantly improving pose ranking. On the model data set however, the ZDOCK baseline performance is considerably worse and DLAB-Re significantly improves ranking performance. **(A, C)**: DLAB-Re ranks the top 500 poses generated by ZDOCK better than ZDOCK, enriching the ratio of pairings with *fnat* > 0.5 poses ranked highly.

Caption continues on next page.

Figure 3.10: continued from previous page

...(**B, D**) Using the DLAB-Re-max score to remove 40%, 60% or 80% of the antibody-antigen pairings respectively can remove antibody-antigen pairings which did not yield high-*fnat* poses. This selects for pairings for which *fnat* > 0.5 poses exist in the top 500 poses generated by ZDOCK (dashed line) and for which DLAB-Re ranks the top 500 poses well (solid line).

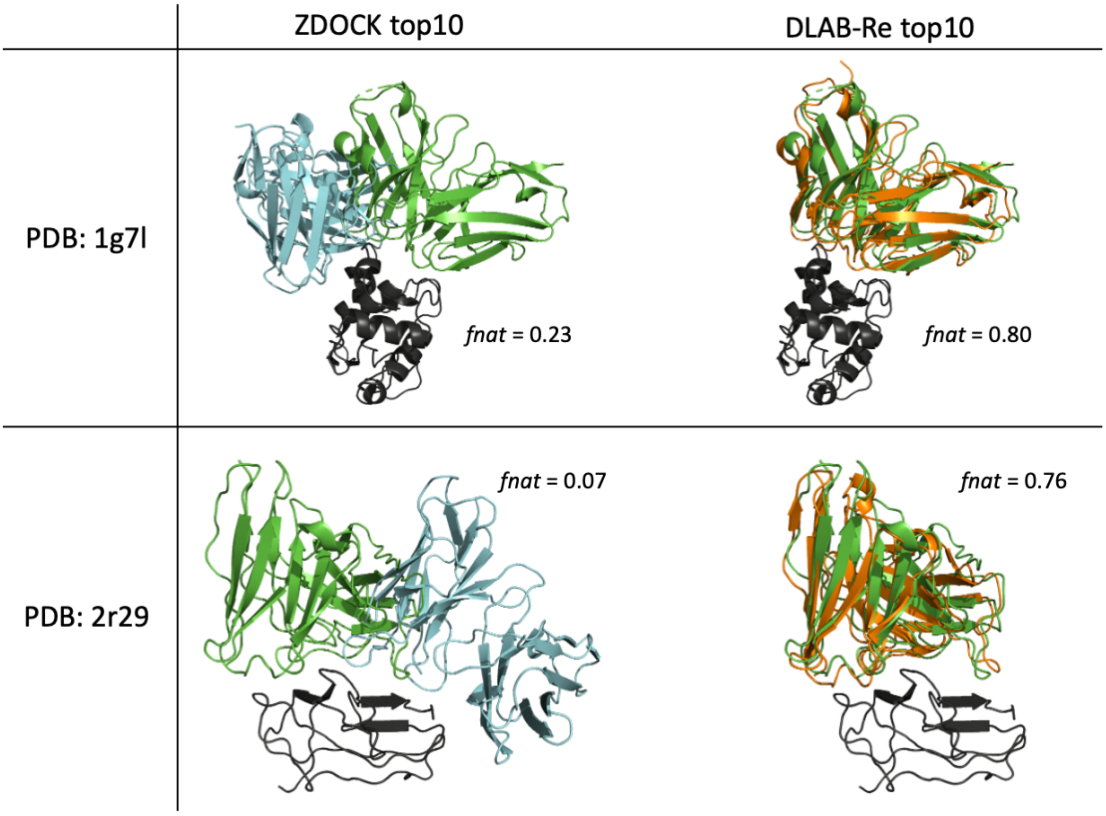

Figure 3.11: Two examples for which DLAB-Re strongly improves pose ranking for modelled antibodies. (Left) Pose with the highest *fnat* in the top ten poses as ranked by ZDOCK. (Right) Pose with the highest *fnat* in the top ten poses as ranked by DLAB-Re. The target antigen is depicted in black, the experimentally determined bound antibody structure in green, the pose with the highest *fnat* in the top ten poses as ranked by ZDOCK in cyan and the pose with the highest *fnat* in the top ten poses as ranked by DLAB-Re in orange.

Side length	Resolution	Accuracy
24.0 Å	0.5 Å	0.129
40.0 Å	0.5 Å	0.207
48.0 Å	1.0 Å	0.354

Table 3.1: Influence of grid dimensions on DLAB-Re performance. Larger grid dimension correlates with higher accuracy on all classes. The values above were derived from training the DLAB-Re model on the crystal structure data set for 100,000 parameter update steps on one of the cross-validation folds. This test was not performed on the final grid definition (r=24 Å), res=0.5 Å, as at the time I assumed this configuration to be too computationally expensive. This was later re-evaluated and correspondingly tested as detailed in Fig. 3.12.

DLAB-Re performance was found to be dependent on both the resolution of the grid in which the protein structure was discretised as well as the size of the grid around the interaction center of the complex structure, with both a larger grid and a smaller resolution increasing the performance of DLAB-Re (see Fig. 3.12, Table 3.1).

3.4.6 DLAB-Re enables identification of successfully docked antibody-antigen pairings

Using the maximum predicted *fnat* score generated by DLAB-Re can allow antibody-antigen pairings where all docked poses are poor to be discarded (see section 3.3.11 and Fig. 3.10 B,D). Choosing thresholds so that 40%, 60% or 80% of pairings respectively are discarded, the remaining pairings are increasingly enriched both in pairings for which the docked poses are ranked well by DLAB-Re as well as in pairings which have at least one pose with a high *fnat* score in the top 500 poses. Discarding 80% of the pairings in the model data set raises

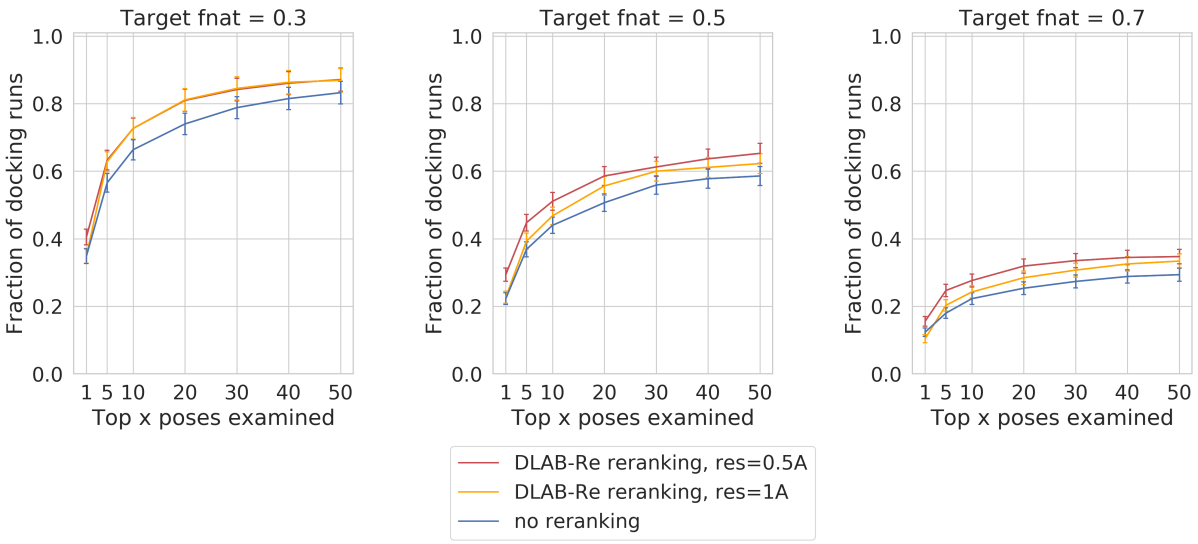

Figure 3.12: Influence of grid resolution on DLAB-Re performance. Models were trained as described in section 3.3.9.3 with either 0.5 Å or 1.0 Angstrom resolution. The 0.5 Å model outperforms the 1.0 Å in early enrichment of poses with *fnat* > 0.5 (Middle) and *fnat* > 0.7 (Right).

the proportion of pairings for which a pose with at least 0.5 *fnat* was ranked by DLAB-Re in the top ten poses from 51% to 84%, meaning that using this thresholding approach eliminated 93% of the antibody-antigen pairings for which DLAB-Re did not manage to rank a pose with at least 0.5 *fnat* in the top ten poses while retaining 33% of the pairings for which it did. This approach therefore enables the identification of antibody-antigen pairings for which accurate complex structures have likely been generated and similarly enables discarding pairings for which the generated complex structures are likely to be of low quality.

Using this approach with the ZDOCK output scores and scores derived from the IPatch reranking approach does not yield the same improvement, only raising the proportion of pairings for which a pose with at least 0.5 *fnat* was ranked by ZDOCK in the top ten poses from 44% to

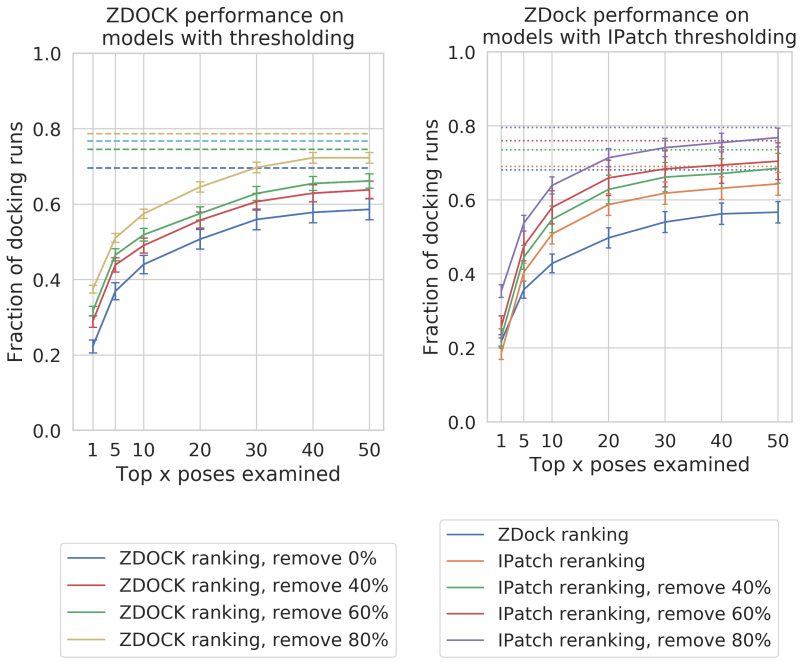

Figure 3.13: Performance of ZDOCK-based and IPatch-based max score thresholding on ranking performance. While higher overall ZDOCK and IPatch scores correlate with improved ranking performance, the effect is much less pronounced than using DLAB-Re-max scores. The solid lines indicate the fraction of antibody-antigen pairings which have at least one pose with *fnat* over 0.5 in their x highest ranked docks, the dotted line indicates the fraction of pairings for which a pose with *fnat* over 0.5 exists in top 500 poses.

57% and 64% respectively (see Fig. 3.13).

3.4.7 A CNN docking rescoring tool trained on crystal structure data does not replicate the DLAB-Re performance

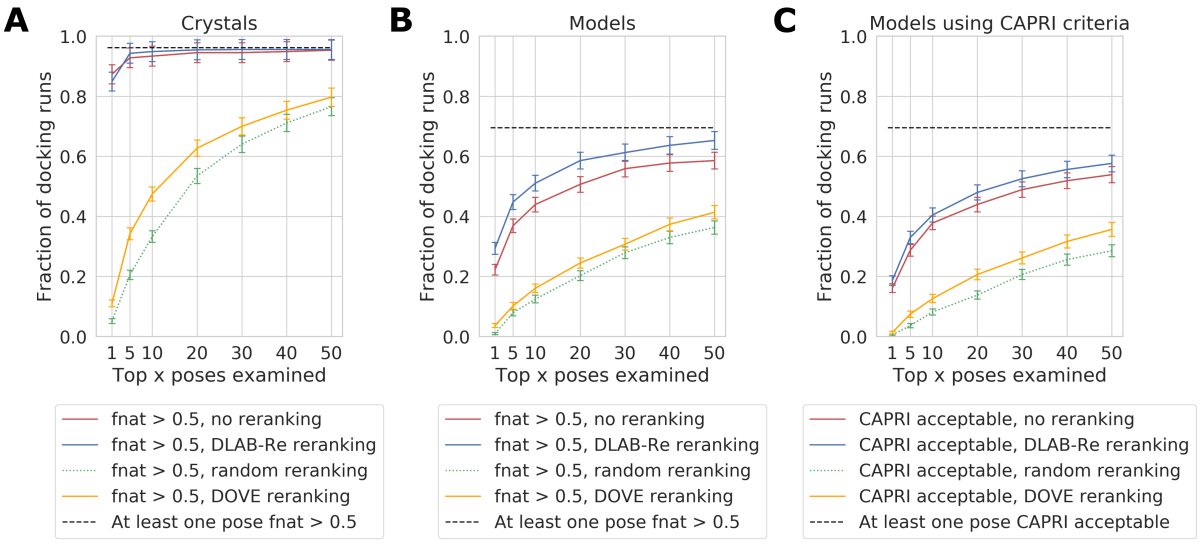

Figure 3.14: Comparison of DLAB-Re performance to DOVE performance. (A, B) Comparison using the pose *fnat* scores on the crystal structure data set (A) and the model data set (B). The fraction of antibody-antigen pairings which have at least one pose with *fnat* over 0.5 in their x highest ranked docks for ZDOCK ranking ("no reranking") as well as DOVE and DLAB-Re reranking is shown in solid lines, the baseline of randomly shuffling the top 500 poses is shown with a dotted green line and the ratio of pairings with a pose with *fnat* over 0.5 is indicated with a dotted black line. (C) The same comparison as (B) using the CAPRI acceptability criterion for each pose. In all three cases (A, B, C), DLAB-Re achieves an improvement over native ZDOCK, while DOVE does not replicate the ranking performance achieved by ZDOCK.

I compared the performance of DLAB-Re with the DOVE tool developed by Wang et al. (2019b). DOVE is a CNN-based docking pose evaluation tool, which is designed to predict

docking pose quality according to CAPRI criteria on crystal structure-based protein-protein docking poses. I used the publicly available ATOM40+GOAP model to test whether this training generalises to the antigen - model antibody docking case. Using the DOVE ATOM40+GOAP score to rerank antibody-antigen docking poses, DOVE performs considerably worse than both ZDOCK and DLAB-Re. This holds for both the crystal structure and the model data set (see Fig. 3.14A and 3.14B).

DOVE is trained to classify docked poses into one of two classes: CAPRI acceptable and not CAPRI acceptable (Wang et al. 2019b; Wodak and Méndez 2004), where CAPRI acceptable poses have *fnat* > 0.1 and interface RMSD < 4 Å or ligand RMSD < 10 Å. Using this classification to evaluate both DOVE and DLAB-Re results, DOVE still performed worse than ZDOCK and DLAB-Re (see Fig. 3.14C). These results highlight the added value from training DLAB-Re both on a domain specific (antibody-antigen) as well as task specific (modelled antibodies) data set.

3.5 Discussion

In this chapter, I have demonstrated the ability of a machine learning augmented antibody modelling and docking pipeline to rapidly generate and assess docked poses of antibody-antigen complexes.

I have shown that DLAB-Re, the first antibody-specific structure-based deep learning approach for docking pose selection and assessment, can be used to improve pose selection in antibody-antigen docking experiments and can enable the identification of antibody-antigen pairings for

which accurate poses have been generated and selected. While DLAB-Re did not significantly outperform the IPatch reranking approach, DLAB-Re is uniquely able to identify pairings for which high *fnat* poses have been generated, identifying a subset of pairings for which a pose with *fnat* > 0.5 is in the top ten poses for 90% of the pairings, which can be used to improve analysis tasks downstream of docking experiments.

I have further shown that the ability to generate accurate complex models of modelled antibodies against cognate antigens using rigid body docking approaches is highly dependent on the capability of modelling tools to generate accurate models of the CDR regions and particularly of the CDR H3 loop. This highlights the particular importance of the ongoing research efforts in the prediction of the H3 loop structure (Abanades et al. 2021; Ruffolo et al. 2020; Weitzner et al. 2017; Marks et al. 2017).

Modelling tools which can achieve higher accuracy in the paratope are available but come at higher computational cost (Weitzner et al. 2017; Marks et al. 2017). RosettaAntibody for example requires on average 1000 CPU hours to generate a final antibody model, compared to 30 CPU seconds for ABodyBuilder, making these methods infeasible for application in the high-throughput use cases on which my research focuses. Recent advances in antibody modelling based on deep learning approaches (e.g. Abanades et al. (2021), see also section 1.2.2) have the potential to enable more accurate antibody modelling in the future while also maintaining the speed advantage of methods like ABodyBuilder.

The antibody-antigen docking pose generation itself could also potentially be improved by using docking tools incorporating flexible docking stages into their algorithm (Ambrosetti et

al. 2019). However, these again come with increased computational cost compared to purely rigid body docking methods, with ZDOCK only requiring five CPU minutes on average to generate and rank 2000 poses for each antibody-antigen pairing. Further, a recent study by Desta et al. (2020) has shown that while flexible docking methods can improve performance on some docking targets, this improvement does not hold across all targets, with rigid body methods outperforming flexible methods on targets that require less conformational change upon binding.

The high-throughput antibody modelling and docking pipeline evaluated in this chapter and augmented by the CNN-based rescoring tool DLAB-Re forms the basis for the virtual screening approaches discussed in detail in the following chapters and presents an easy to implement tool for the fast and accurate analysis of large areas of the antibody-antigen interaction space.

4 | Classification of antibody-antigen binding

Contents

4.1 Chapter motivation

Having improved the ranking of antibody-antigen docking poses with the DLAB-Re tool as detailed in chapter 3, the remainder of my DPhil research aimed at developing structure-based machine learning tools for antibody virtual screening. This chapter details the development of DLAB-VS, the first machine learning-based antibody virtual screening tool using structural information.

The work detailed in this chapter forms part of "DLAB: deep learning methods for structure-based virtual screening of antibodies" (Schneider et al. 2021b) and as such, parts of this chapter are adapted from this paper.

4.2 Introduction

In order to develop a successful therapeutic antibody, several features have to be optimised. Antibodies need to bind with high efficacy, specificity, and affinity to the target of interest, while at the same time avoiding an immune reaction in the patient (Suscovich and Alter 2014) as well as avoiding properties that lead to poor developability, such as self-association, viscosity, or immunogenicity (Raybould et al. 2019b). To achieve these goals, large-scale experimental screens are usually employed in the pre-clinical stages of antibody-drug development (Suscovich and Alter 2014; Almagro et al. 2018).

As described in the introduction (see section 1.4.1), initial leads for therapeutic antibodies are generated using either *in-vitro* approaches using display platforms or *in-vivo* approaches using

transgenic animals or patient derived data. Further improvement of these initial hits is then achieved through affinity maturation, either via generation and screening of further, hit-based mutagenesis libraries or via rational engineering. High-affinity antibodies generated in this way can be further engineered to achieve desirable properties for antibody therapeutics, for example changes to the constant region to modulate the effector functions as well as *in-vivo* half-life and the improvement of developability (Chiu and Gilliland 2016).

These experimental methods are often effective at generating high-affinity antibodies for downstream development but are cost- and time-intensive. Furthermore, they do not as standard generate insights into the binding mode of the generated antibodies.

This development pipeline is now commonly supplemented using in-silico methods (Raybould et al. 2019a; Norman et al. 2020), particularly once initial binding candidates have been identified. Computational tools have been used to rationally engineer antibody binding sites in several studies (e.g. Lapidoth et al. (2015), Adolf-Bryfogle et al. (2018), and Chowdhury et al. (2018)). For a more detailed overview of computational tools employed in antibody engineering, see section 1.4.4.

As described in the introduction (see section 1.4.4), machine learning approaches in the field of antibody therapeutics discovery have tended to focus on sequence rather than on structure as input (e.g. Olimpieri et al. (2013), Bujotzek et al. (2015), Liberis et al. (2018), and Mason et al. (2021)). These machine learning approaches have been shown to be highly efficient for prediction tasks which depend only on the antibody (rather than on the antibody-antigen interface) or are specific to one antigen.

Sequence-based machine learning approaches have yet to provide generalisable predictions across different antigens in one model. An approach in this direction has recently been explored by Akbar et al. (2021a), who generated machine-learning accessible descriptions of the binding interface using structure-derived interaction motifs. Further, structure-based approaches have recently been used for the prediction of antibody and antigen interface residues (e.g. Pittala and Bailey-Kellogg (2020), see also section 1.6.2.2), demonstrating the ability of machine learning approaches to successfully utilise structural information derived from both the antibody and the antigen.

In this chapter, I describe a structure-based deep learning approach for early-stage virtual screening of antibody therapeutics, when an epitope target of interest is known but no viable hit antibodies have yet been identified. My approach is able to make generalisable predictions across different antigens. Adopting the same three-dimensional gridding method used in chapter 3 and a similar convolutional neural network trained on docked poses of modelled antibody structures in complex with antigen epitopes generated using the improved antibody docking pipeline I described in the previous chapter, I built a **D**eep **L**earning approach for **A**nti**B**ody **v**irtual **s**creening (**DLAB-VS**). Using DLAB-VS, I demonstrate that machine learning approaches using structural antibody data can enable large scale computational screening in the antibody development pipeline.

4.3 Methods

4.3.1 Model data set and crystal structure data set

The model data set and the crystal structure data set used to train and validate the DLAB-VS models described in this chapter were created as described in chapter 3 in section 3.3.1. Clustered cross-validation splits were generated as described in section 3.3.7.

4.3.2 Generation of binding and non-binding antibody-antigen pairings

I considered every antibody-antigen pair in the crystal structure data set as a binding pair. Since antibodies bind with high specificity, I generated non-binding antibody-antigen pairs by randomly sampling 50 non-cognate antibodies per antigen from the crystal structure/model antibody data set. My definition of non-cognate required the non-cognate antibodies to share less than 90% of their CDR sequence with the binding antibody. These antibody-antigen pairings could be considered non-cognate, as the CDR sequence is the primary determinant of antibody-antigen binding (see section 1.1.1). Since the median combined CDR length of full-Fv antibodies has been estimated to be 49 amino acids (Raybould et al. 2019b), this cutoff meant that antibodies considered non-cognate differed on average in at least five CDR amino acid positions from the cognate antibody, making it highly unlikely for those antibody-antigen pairings to be false negatives.

4.3.3 Antibody-antigen pose generation

4.3.3.1 ZDOCK antibody-antigen docking

I generated docked poses of binding and non-binding antibody-antigen pairs using the docking pipeline described in chapter 3 in section 3.3.4.

4.3.3.2 Idealised pose generation through superimposition and local optimisation

In order to investigate the influence of model quality on classification performance, I used an idealised docking approach, where the experimentally determined antibody-antigen interaction pose was used to guide the pose generation for the model data set. For each antibody-antigen pair (cognate and non-cognate), I created an initial pose by superimposing the framework of the modelled antibody on the experimentally determined framework coordinates of the cognate antibody in complex with the target antigen using the Biopython package (Cock et al. 2009). This initial pose of the modelled (cognate or non-cognate) antibody in complex with the antigen was refined by repacking the side chains and then performing local high resolution energy minimization using the Rosetta docking tool (Adolf-Bryfogle et al. 2018).

4.3.3.3 Pose voxelisation

Antibody-antigen poses were voxelised using the strategy described in section 3.3.8 (the libmolgrid voxelisation and atom-typing pipeline), generating grids of side lengths $24\,\text{Å}$, $32\,\text{Å}$, $40\,\text{Å}$ and $48\,\text{Å}$ as well as $1.0\,\text{Å}$ and $0.5\,\text{Å}$ resolution. I compared this typing approach to a simplified atom typing approach using only the heavy atom element type to generate four in-

put channels (C, O, N, S). In the following, I refer to this second scheme as the CONS typing scheme.

4.3.3.4 DLAB-Re pose reranking

I used the DLAB-Re models generated in chapter 3 to rerank the top 500 poses for all antibody-antigen pairings used during training and evaluation of DLAB-VS. I used the model weights derived during cross-validated training. In the case of cognate pairings, I used the DLAB-Re model not trained on the pairing. In the case of non-cognate pairings, I used the DLAB-Re model not trained on the antigen.

As introduced in the previous chapter, to identify antibody-antigen pairings with low quality docking poses, I determined the highest DLAB-Re score given to any of the top 500 ZDOCK poses for each antibody-antigen pairing. This score (DLAB-Re-max) was used to discard particular pairings by ranking all pairings by their DLAB-Re-max score and discarding the bottom 80%.

4.3.3.5 Antibody model quality determination and docked pose quality determination

I calculated the RMSD over different regions of the antibody models as described in section 3.3.3. For cognate antibody-antigen pairings, I calculated the *fnat* score as described in section 3.3.5.

4.3.4 ZDOCK-based virtual screening

For the ZDOCK based classifier, the ZDOCK score of the top-ranked binding pose was used to rank the antibodies docked against a particular target antigen.

4.3.5 Virtual screening model development

4.3.5.1 Overview

The goal of virtual antibody screening is to discern binding antibodies against a given epitope from a pool of candidate antibodies. To generate a classification model able to accomplish this task, I trained several CNN models (architectures depicted in Fig. 4.1) that classify individual antibody-antigen complexes as binder or non-binder.

4.3.5.2 Model architectures

I tested two model architectures during the development of DLAB-VS (see Fig. 4.1). The first architecture resembles the model architecture used first by Ragoza et al. (2017) for small molecule virtual screening and consists of three 3D convolutional layers prepended by max-pooling operations, followed by flattening of the pose feature tensor and classifying with a fully connected layer followed by sigmoid activation function. In the following, I refer to this architecture as the LeNet-like model due to its similarity to the LeNet CNN architecture (Lecun et al. 1998). The second architecture was adapted from Imrie et al. (2018) and employs Dense-Blocks to facilitate efficient information flow through the network at larger overall network depths. In the following, I refer to this architecture as the DenseNet-like model, in reference to

Training set	binder samples	non-binder samples
1	+ve DLAB-Re Top 1s	-ve DLAB-Re top 1s
2	+ve best with *fnat* >= 0.7	-ve DLAB-Re top 1s
3	+ve best with *fnat* >= 0.7	-ve DLAB-Re top 1s
	clonotype expansion of binders	
4	+ve all with *fnat* >= 0.7	-ve DLAB-Re top 1s
		+ve with *fnat* < 0.1
5	+ve all with *fnat* >= 0.5	-ve DLAB-Re top 1s
		+ve with *fnat* < 0.1
6	+ve all with *fnat* >= 0.7	-ve DLAB-Re top 1s
	clonotype expansion of binders	+ve with *fnat* < 0.1

Table 4.1: Overview of training set compositions for DLAB-VS training on the model data set. +ve refers to binding antibody-antigen pairs, -ve refers to non-binding pairs. Clonotype expansion of binders refers to the expansion strategy detailed in algorithm 1.

the paper which first introduced DenseBlocks (Huang et al. 2016).

4.3.5.3 Training set compositions

During the development of DLAB-VS, I tested several approaches to select binder and non-binder samples for the training set (see Table 4.1). For non-binding antibody-antigen pairs, the highest ranked pose after DLAB-Re rescoring was selected as a non-binding pose. For binding pairs, I initially selected the highest ranked pose after DLAB-Re rescoring as a binding pose (training set composition 1). Following the approach taken by Scantlebury et al. (2020), I also tested several training set compositions where I selected either the best docked pose with *fnat* > 0.7 (where available) as a binding pose (training set composition 2, 3) or up to 50 poses

Figure 4.1: The DLAB-VS architectures used in the final ensemble of DLAB-VS models. Both architectures output a binary binder/non-binder classification for an antibody/antigen pairing. (**Top**) A 3-layer CNN (referred to as LeNet-like) followed by a fully connected layer and a sigmoid activation function is used to classify input structures as either binder or non-binder. (**Bottom**) A CNN architecture consisting of a single convolutional layer followed by 3 DenseBlocks, using the same output layer as above, referred to in the text as DenseNet-like.

with *fnat* > 0.7 or *fnat* > 0.5 as binders (training set compositions 4, 5, 6). In those training set compositions, I also selected five poses of binding pairs with *fnat* < 0.1 as non-binders in order to force the networks to learn from the interaction between antibody and antigen rather than just memorising the interacting pairings by providing for the same antibody-antigen pairing both good and bad binding poses.

Further, I used a clonotyping strategy to increase the amount of binding antibody-antigen pairs in the model data set (training set compositions 4, 6). Here I identified naturally occurring antibodies which are sufficiently similar to a confirmed binding antibody to treat them as binders during training. For this purpose, for each antibody for which binding poses were added to the training set, I searched the observed antibody space (OAS) database (Kovaltsuk et al. (2018), see also section 1.2.2.1), downloaded on 21/01/2020, with the following algorithm and added identified antibody matches to the training dataset as binders:

Algorithm 1 OAS clonotypal expansion.

query antibody heavy chain: Q_H
query antibody light chain: Q_L
query paratope residues: $P(Q_H) = \{p_1, p_2, ..., p_i\}$
for Entry E in OAS **do**
  if VJ(E) == VJ(Q_H) **then** ▷ VJ-Gene assignment via ANARCI
    if len(CDR3(Q_H)) == len(CDR3(E)) **then**
      if $|P(Q_H) \cap P(E)| > 0.9 \times |P(Q_H)|$ **then**
        <Generate model M of Q_L with E>
        if RMSD(M, Q) < 1 Å **then**
          <Add model to training set as binder>
        end if
      end if
    end if
  end if
end for

For each query heavy chain Q_H, I identified OAS entries which shared the V and J gene assign-

ment of Q_H as well as having the same length CDR H3 loop and at least 90% overlap in the paratope residues. I generated models of those entries paired with the query light chain using ABodyBuilder. If the RMSD between the new model and the crystal structure of the query antibody was less than 1 Å across each CDR loop, the new model was added to the training set as a binder to the cognate antigen of the query antibody (see Algorithm 1).

4.3.5.4 Model training

Data augmentation was performed in the same manner as for DLAB-Re, see section 3.3.9.3. Models were trained for 50,000 parameter update steps using the rectified Adam optimiser. A validation set comprising 10% of the total data set was created using the same CD-HIT clustering as for the training set creation. The validation set was used to select a snapshot of the model during training by choosing the model snapshot with the highest average precision on the validation set. To counteract class imbalance, binder and non-binder poses were sampled so that each batch contained equal numbers of both classes.

4.3.5.5 Pose score combination

At prediction time, three pose score combination approaches were used: the output score for each antibody-antigen pairing was either set as the DLAB-VS score of the top ranked pose or as the average or the maximum value of the top ten poses. As described above, I used DLAB-Re reranked poses for this purpose.

4.3.5.6 Model ensemble creation

In order to generate more robust virtual screening models, I generated model ensembles. For each cross-validation fold and network architecture, I used two different train/validation set splits to train an ensemble of four models per fold. I calculated the output scores for each antibody-antigen pairing by averaging the output scores of the ensemble member models.

In order to create an ensemble model comprised of DLAB-VS and ZDOCK, the DLAB-VS output scores and the ZDOCK output score were normalised per antigen target via minmax scaling and averaged to arrive at the final scores for each target antigen. This was necessary since the DLAB-VS output scores are bounded in [0,1], whereas the ZDOCK score is unbounded.

4.3.6 Additional data sets

4.3.6.1 Post-snapshot model data set

To create an unseen test set, which was not used at any point during model choice and hyperparameter optimisation, I used SAbDab entries deposited between 20/12/2018 (i.e. after the date of the snapshot used for the training data set described in chapter 3) and 20/05/2020 to create an unseen test set. This data set, referred to in the following as the post-snapshot model data set, contained 222 antibody/protein antigen complexes, which formed 173 CDR clusters after clustering the CDR-sequences using CD-HIT at 90% identity. On this test set, I performed modelling, docking, rescoring and binder classification as described above, using the models trained on the model data set. For calculation of the ensemble DLAB-VS score, I combined all 40 previously trained models into one ensemble. The PDB accession codes for this data set can

be found in Appendix Table B.1.

4.3.6.2 Epitope cluster test set

To create a series of test sets closer to a real usage scenario for an antibody virtual screening tool, I identified epitope clusters in the model data set (epitopes with a large number of known, distinct antibody binders).

To reduce the amount of pairwise comparisons necessary, I clustered antigens using CD-HIT at 50% sequence ID. For each cluster, I calculated pairwise epitope overlap using Biopython (Cock et al. 2009) and used greedy clustering to identify antibodies which bound to epitopes with at least 50% overlap. I identified two clusters (binding against lysozyme and HIV GP-120) with at least ten cluster members (Fig. 4.16 and Tables B.2 and B.3). For each of those clusters, I created a test case comprised of the binding antibody-antigen pairings of the cluster and all non-binding antibody-antigen pairings in the model data set containing the cluster epitope.

4.3.6.3 Coronavirus test set

An additional test set was created using publicly available COVID-19 patient antibody sequencing data (Galson et al. 2020) alongside confirmed binder sequences against SARS-CoV-1 (Sui et al. 2008) to create a potential realistic test case, referred to in the following as the SARS-CoV model set. Models of the anti-SARS-CoV-1 antibody 80R (Sui et al. 2005) as well as five confirmed binding light chain variants of 80R and 13 clonotype matches to 80R (same VJ gene assignment via ANARCI (Dunbar and Deane 2015) and > 80% CDR H3 sequence identity) derived from COVID-19 patient antibody repertoires were docked against the 80R epitope on

the SARS-CoV-1 RBD as well as the corresponding area on the SARS-CoV-2 RBD. To generate negative samples, two sets of 300 samples from an antibody model library (AML) derived from the observable antibody space (OAS) database (Kovaltsuk et al. 2018) were also docked against the two epitopes. The two sets differed in their H3 length distribution: in one set, AML antibodies of any H3 length were randomly selected, while in the other set, the AML antibodies had to match the 80R H3 length +/- 1 amino acid to avoid the trivial criterion of H3 length clustering. I performed docking, rescoring and binder classification as described in section 4.3.3.4 and 4.3.5, using the large ensemble of 40 models since neither 80R nor any antibody with a CDR sequence with > 80% sequence identity to 80R was included in the original training set.

4.3.6.4 Variant binding test set

I created a second SARS-CoV-2 data set (the SARS-CoV-2 variant data set) by extracting all antibodies from the Coronavirus Antibody Database (CoV-AbDab) (Raybould et al. 2021a) which were confirmed to bind to the SARS-CoV-2 wild-type RBD and also confirmed not to bind to at least one SARS-CoV-2 variant and for which an experimentally determined complex structure was available from which the epitope could be determined. ABodyBuilder was used to model the antibodies as described in section 3.3.2. The structural models of the variant antigens were created using Foldx5 (Schymkowitz et al. 2005) and the PDB files listed in Table B.4 as templates and for docking purposes, the epitope definition from the templates was copied onto the variant models. I then docked each antibody model, defining the paratope as described in section 3.3.4.2, against its epitope on both the wild-type RBD and the confirmed non-cognate variant RBDs and performed rescoring and binder classification as described above, using the

40-model ensemble. As for most of the epitopes in this set, only one antibody was docked, score normalisation was performed over the entire dataset instead of on a per-epitope basis.

4.4 Results

4.4.1 DLAB-VS and ZDOCK are able to rank crystal structures accurately

Virtual screening for antibody discovery aims to find binding antibodies for a given epitope from a large set of potential binders. To demonstrate the feasibility of my machine-learning based approach DLAB-VS, I first attempted to retrieve correct binders from the crystal structure data set by docking both the cognate antibody crystal structure as well as 50 non-cognate antibody crystal structures against each antigen crystal structure in the data set. Due to the high-quality of experimentally determined structures, this task is considerably easier than the more realistic case of using computationally predicted structures, thus providing a good initial proof-of-principle. Using ZDOCK scores to classify antibody-antigen pairings, the cognate binder is found in the top 2% (i.e. top ranked) of antibodies for 49.7% of antigen targets, and in the top 10% in 65.6% of antigen targets, outperforming the random baseline. Training a convolutional neural network model (using the LeNet-like architecture and training set composition 1, see Fig. 4.1 and Table 4.1), termed DLAB-VS, on the crystal structure data set, the cognate binder is found in the top 2% and top 10% of antibodies for 62.6% and 72.7% of antigen

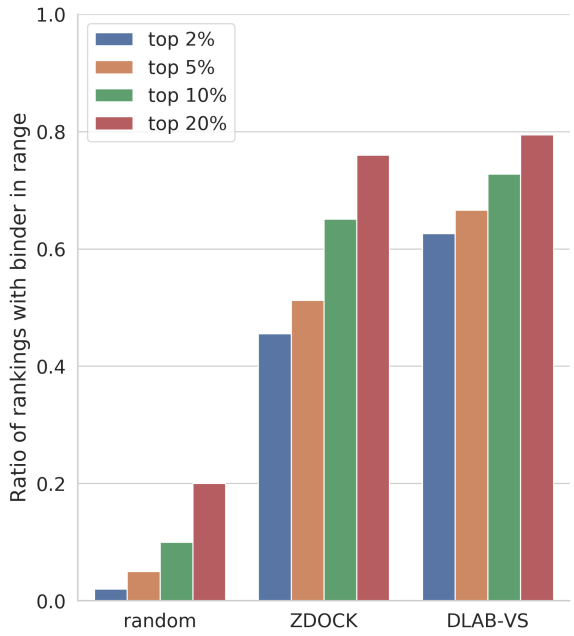

Figure 4.2: Binder classification performance of ZDOCK and DLAB-VS on crystal structure data. The ratio of antigen targets for which the binding antibody was ranked in the top 2%, top 5%, top 10% and top 20% respectively is shown. Both ZDOCK and DLAB-VS achieve high classification performance on this data set.

targets respectively (see Fig. 4.2[1]). These results demonstrate that using convolutional neural networks on the antibody binder classification task is a valid approach and can outperform classical approaches.

[1]The data presented in this figure as well as similar data shown in this chapter and chapter 5 was generated by aggregating the results of several cross-validation folds. In future work, in order to be able to provide confidence metrics, summary statistics could instead be calculated across the training folds.

4.4.2 Naive DLAB-VS alone performs badly when trained on models using a non-curated training set

I next assessed the performance of DLAB-VS trained on the model data set using the same training set composition employed for the crystal structure data set (using the highest ranked pose for each positive and negative pairing in the training set, training set composition 1 in Table 4.1). Using this training data, DLAB-VS failed to replicate the classification performance achieved on the crystal structure data set, just slightly outperforming the random baseline with the binder in the top 2% and top 10% of antibodies for 3.5% and 12.1% of antigen targets respectively and an overall average precision of 0.021 (where the expectation value is 0.02) (see Fig. 4.3).

It should be noted that I used the Parapred definition of the paratope (see section 3.3.4.2) for the data set creation in this section and the following section 4.4.3. As I detailed in the previous chapter (see section 3.4.3), I later switched to the IMGT+2 definition of the paratope, which is therefore used for all results starting with section 4.4.5. This switch only resulted in a small improvement of the quality of docked poses and therefore does not affect the validity of the results detailed here.

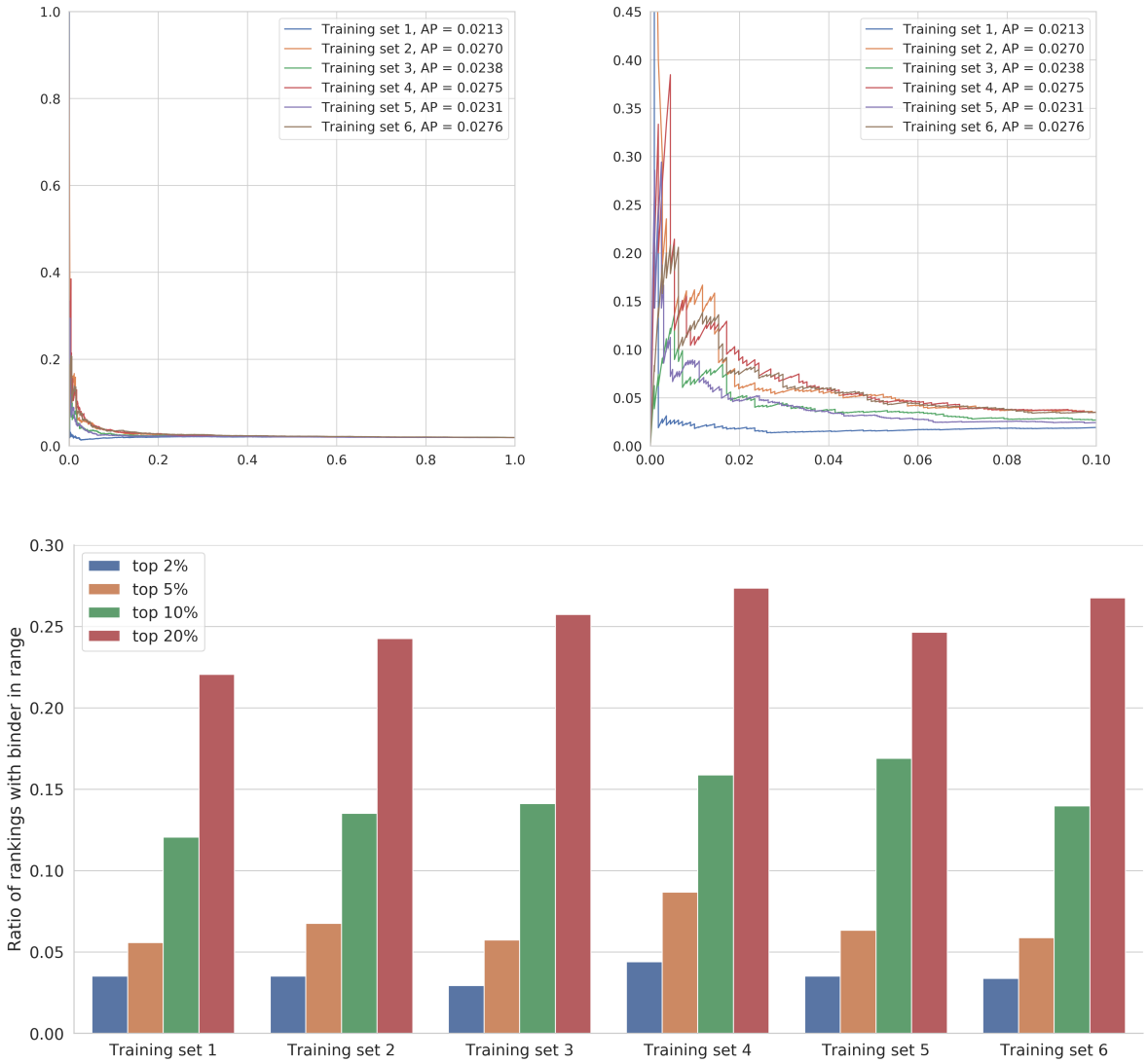

Figure 4.3: Influence of different training set compositions on DLAB-VS performance on the model data set. **(Top)** Precision-Recall curves resulting from different training set compositions for the model data set. The right graph is a magnified view of the left graph. The training set names (1-6) quoted refer to the sets described in Table 4.1. On the naive training set 1, performance is approximately random, with average precision (AP) ≈ 0.02. Changes to the training set composition as detailed above enable improvements over that performance. **(Bottom)** Binder ranking performance per antigen target using the same data sets...

Caption continues on next page.

Figure 4.3: continued from previous page

...For each training set composition, the ratio of antigen targets for which the binding antibody was ranked in the top 2%, top 5%, top 10% and top 20% respectively is shown. Training set 4 confers an advantage in early enrichment of binders.

4.4.3 Using different training set compositions improves the performance of DLAB-VS on the model data set

There are several aspects of the model data set which hamper the ability of machine learning models to accurately classify antibody-antigen binding.

Firstly, for a large proportion of model antibody structures, the docking pipeline does not yield high-quality complex structures (see Fig. 3.10), even after the improvements to the docking pipeline made in the previous chapter. Following the approach by Scantlebury et al. (2020), I implemented several training set composition schemes based on the accuracy of the docked poses, determined by *fnat*, as detailed in section 4.3.5.3.

Secondly, the relative sparsity of available antibody-antigen structural data limits the number of positive samples in the training set (additional non-binding decoys on the other hand can be generated easily). To address this, I implemented a clonotyping approach for the extension of the positive samples in the training set, as detailed in section 4.3.5.3. All training set compositions tested are listed in Table 4.1.

The training approach derived from Scantlebury et al. (2020) (used in training sets 4, 5 and 6) increased the performance of DLAB-VS on the model data set. Training set 4, which labelled

as binders up to 50 docked poses with *fnat* > 0.7 for each cognate antibody-antigen pairing and as non-binders 10 docked poses with *fnat* < 0.1 for each cognate pairing in addition to the top ranked pose for each non-cognate pairing, increased the overall average precision from 0.021 to 0.028 and the ratio of binders in the top 2% and top 10% of antibodies against each antigen from 3.5% and 12.1% to 4.4% and 15.9% respectively. Using a lower *fnat* threshold of 0.5 (training set 5) or only the best docked pose (rather than multiple docked poses meeting the *fnat* threshold) (training set 2), resulted in smaller improvements than training set 4.

The clonotyping based expansion of the antibody-antigen pairings labelled as binders (training sets 3 and 6) did not offer any further improvement over training sets 1 and 4 and in fact appeared to decrease the performance of the trained model slightly, indicating that this process did not add informative binding antibody-antigen pairings to the training set (and in fact might have added noise to the training set).

In all following experiments, training sets were therefore generated according to training set composition 4.

4.4.4 Exploration of different model parameters on idealised data shows the importance of high resolution and high content input data

Apart from training set composition, a secondary set of hyperparameters not directly related to the model architecture are the parameters used to generate the three-dimensional, voxelised grids used as input for the CNN models used in this thesis. To rapidly explore the importance of these hyperparameters and decide on the best set to use in following experiments, I ran a series

of experiments on an idealised model data set, in which the training and test set were composed of the best docked pose (highest *fnat*) for each cognate antibody-antigen pairing and the highest ranked pose for each non-cognate pairing. Using this dataset, I tested two different atom typing schemes, four different sizes of the grid centered on the antibody-antigen interaction interface and two different training durations as described in sections 4.3.3.3 and 4.3.5.4, assessing the performance of the trained model using the average precision achieved on the test set.

Finer resolution of the input grid correlated with improved performance, increasing the average precision by 18.3% on average. The more granular libmolgrid typing scheme improved the average precision over the CONS typing scheme by 38.1% on average. Doubling the training time increased the average precision by 23.1% on average (see Fig. 4.4). Therefore, I used a grid size of $48\,\text{Å}$ with a resolution of $0.5\,\text{Å}$ alongside a training time of 50,000 parameter update steps in all following experiments.

4.4.5 Pose ensembles slightly improve early binder enrichment

In the previous chapter, I demonstrated that docking modelled antibodies against experimentally determined antigen structures yields a small ratio of antibody-antigen pairings for which the top ranked pose is well docked (29.4% of top ranked poses after DLAB-Re rescoring have *fnat* > 0.5, see Fig. 3.10), while the ratio of pairings for which a well docked poses is present in the top ten ranked poses is significantly larger (51.1% of antibody-antigen pairings, see Fig. 3.10). Therefore, I investigated whether using an ensemble of docked poses at test time could improve the ability of DLAB-VS to predict antibody-antigen binding. I compared the performance of using the DLAB-VS output on the top ranked docked pose with the average as

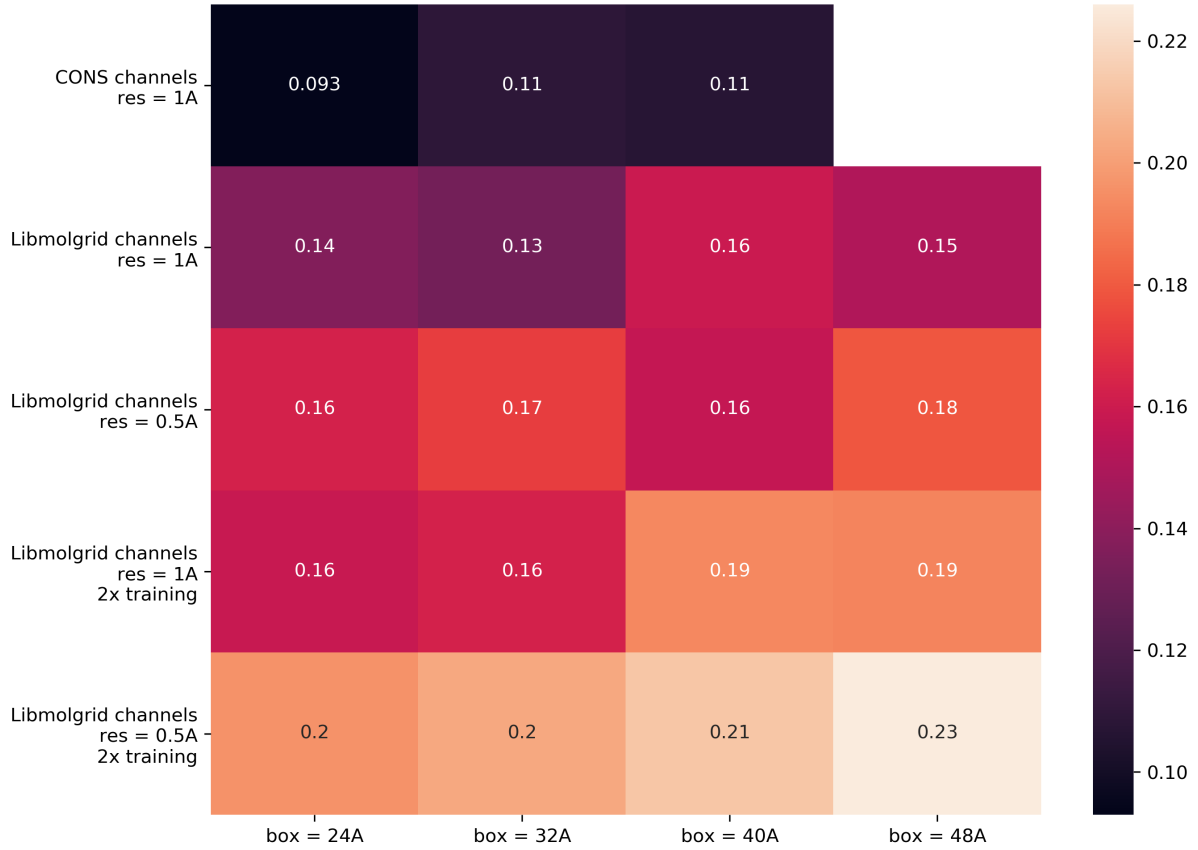

Figure 4.4: Heat map detailing the influence of different hyperparameters on the average precision achieved on the model data set using the best docked pose for cognate antibody-antigen pairings described above. For each hyperparameter combination, I quote the average precision achieved on one test dataset fold. CONS channels and libmolgrid channels refers to the typing schemes described in section 4.3.3.3. Finer resolution of the input grid, a larger grid size and finer channel typing all correlate with higher average precision. Due to time constraints and the low performance of the CONS channel typing method, I did not evaluate the combination CONS typing, 48 Å box size and 1 Å resolution.

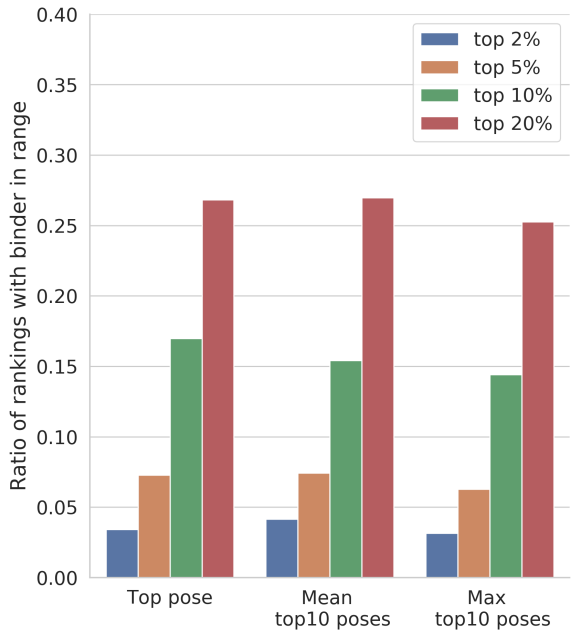

Figure 4.5: Influence of using pose ensembles on DLAB-VS performance. I compare the performance of DLAB-VS, training on training set 4, using just the DLAB-VS score of the top ranked pose (as ranked by DLAB-Re), the mean and the maximum of the 10 top ranked poses. Using the mean of the top 10 poses achieves slightly higher early enrichment of binders.

well as the maximum of the DLAB-VS output on the top 10 ranked poses (see Fig. 4.5).

Using the average score across the top 10 poses resulted in the binding antibody being ranked in the top 2% for 4.1% of antigen targets, as opposed to 3.1% for the maximum of the top 10 poses and 3.4% for the top ranked pose only (see Fig. 4.5). I therefore used the average scoring scheme in the following experiments in this chapter.

4.4.6 Model ensembles improve prediction

A commonly employed approach to improve the performance of machine learning models is the use of ensemble models (Maclin and Opitz 1999). Ensemble model methods rely on gener-

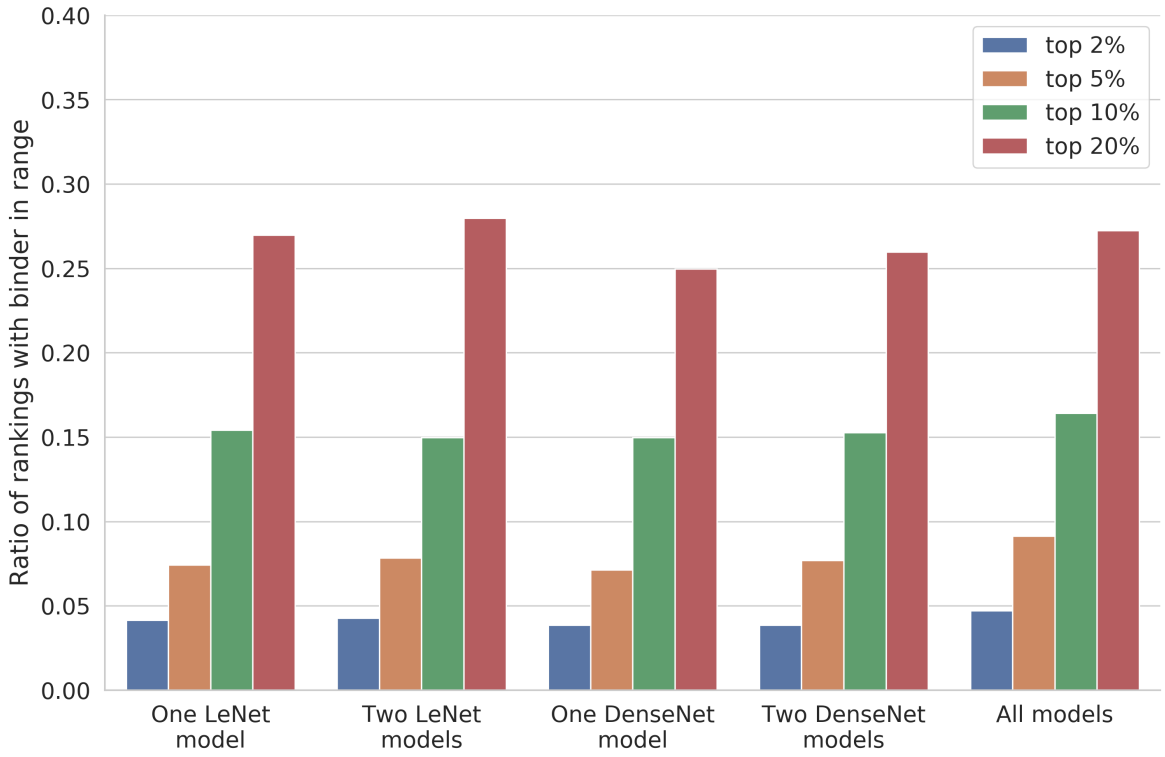

Figure 4.6: Performance of different ensembles of DLAB-VS models. The ensembles are composed of either one model, two models of the same architecture or four models, two of the LeNet-like architecture, two of the DenseNet-like architecture. While DenseNet-like models on their own perform slightly worse than LeNet-like models, combining both yields improvements over just using one model type.

ating several models with (partially) uncorrelated errors and combining the generated models in order to reduce overall model variance. The simplest approach to this end is bagging, in which component models of the ensemble are trained independently and their scores are averaged at prediction time. Models with uncorrelated errors can be generated by varying training hyperparameters, model architecture or training set composition.

In order to improve the performance of the DLAB-VS model, I generated several bagging ensembles, varying both training set composition and model architecture (see Fig. 4.1). For each cross-validation fold, I generated two distinct train/validation set splits and used each to train a LeNet-like and a DenseNet-like model. The four models trained on each fold were combined in three ensembles (2x LeNet-like, 2x DenseNet-like, all models) and compared to the performance of one model of each architecture (see Fig. 4.6). Ensemble models performed better than each component model, with the ensemble combining all four models achieving the best performance, ranking the binding antibody in the top 2% and top 10% for 4.7% and 16.4% of antigen targets respectively, up from 4.1% and 15.4% respectively for the non-ensemble approach.

While the DenseNet-like model performed worse than the simpler LeNet-like model both as a single model and in a two-model ensemble (see Fig. 4.6), the errors between the two model architectures were partially uncorrelated (see Fig. 4.7), and the addition of the two DenseNet models to the ensemble therefore enabled an increase in the overall performance.

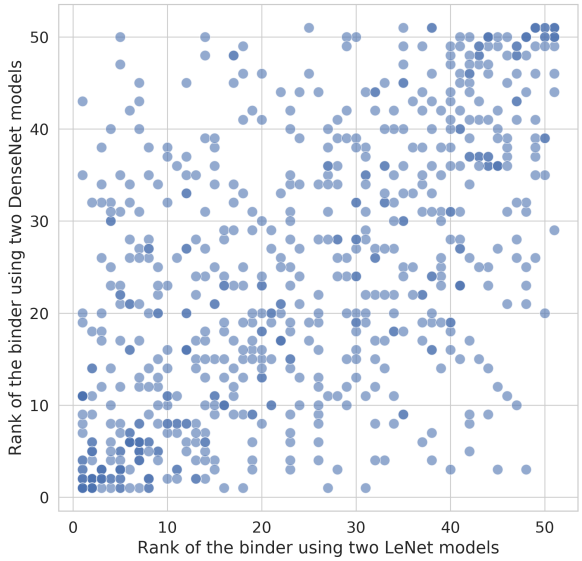

Figure 4.7: Scatter plot of ranking performance of DenseNet and LeNet DLAB-VS models.

4.4.7 Combining the DLAB-VS scores with ZDOCK scores improves prediction

I further investigated whether generating a bagging ensemble combining the scores generated by ZDOCK for each antibody-antigen pairing with the classification score generated by the DLAB-VS model could improve the overall classification performance. ZDOCK achieved similar classification performance to the DLAB-VS ensemble model for top 2% and top 5% performance (see Fig. 4.8), however ZDOCK ranking and DLAB-VS ranking showed little correlation (see Fig. 4.9), thus enabling an improvement in classification performance through bagging: the DLAB-VS+ZDOCK ensemble model ranked the binding antibody in the top 2% and top 10% for 6.4% and 19.7% of antigen targets respectively, up from 4.7% and 16.4% re-

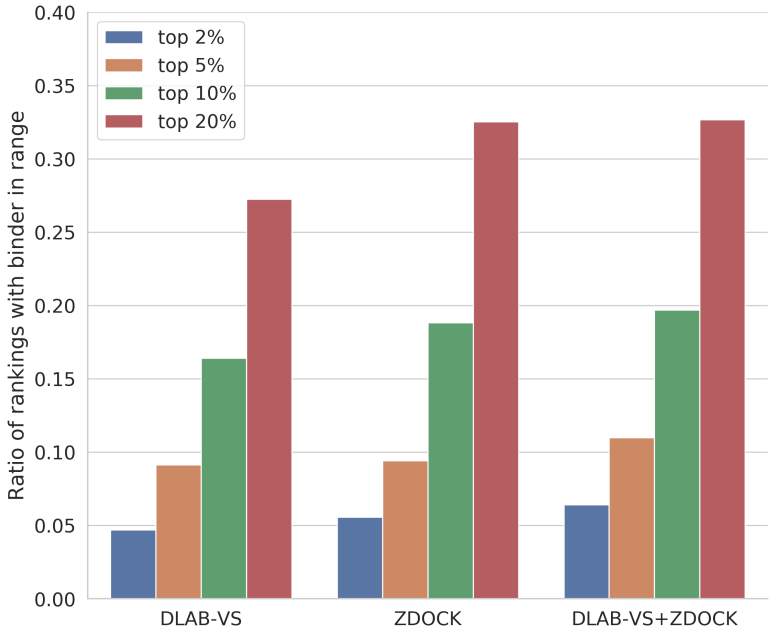

Figure 4.8: Comparison of the binder classification performance of DLAB-VS, ZDOCK, and DLAB-VS+ZDOCK. Combining the two models gives a gain in classification performance over either of the component models in binders ranked in the top 2% and top 5% of antibodies docked against each antigen.

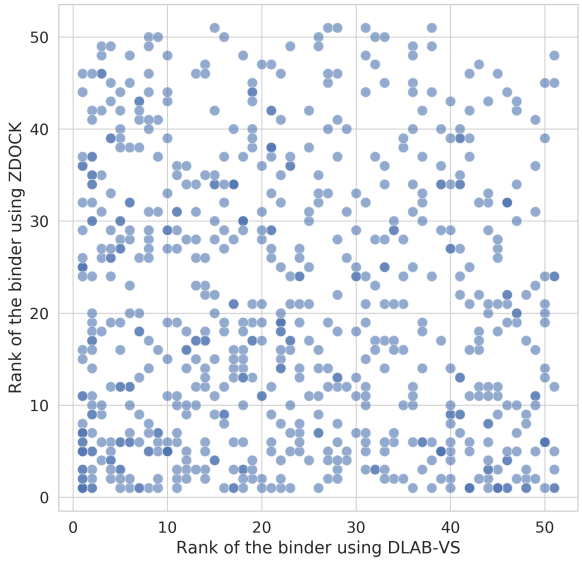

Figure 4.9: Scatter plot of ranking performance of DLAB-VS and ZDOCK.

spectively for DLAB-VS ensemble. In the following, I therefore used the DLAB-VS+ZDOCK ensemble.

4.4.8 Ranking performance depends strongly on model quality and docking quality

The performance of binder classification tools differs between the crystal structure and the model data set. In order to understand how strongly this difference is driven by model quality and docking quality respectively, I stratified the performance of the DLAB-VS+ZDOCK model dependent on the all-CDR RMSD (calculated as described in section 3.3.3) of the modelled antibody as well as the docking quality achieved by ZDOCK on the binding antibody-antigen pairings (see Fig. 4.10).

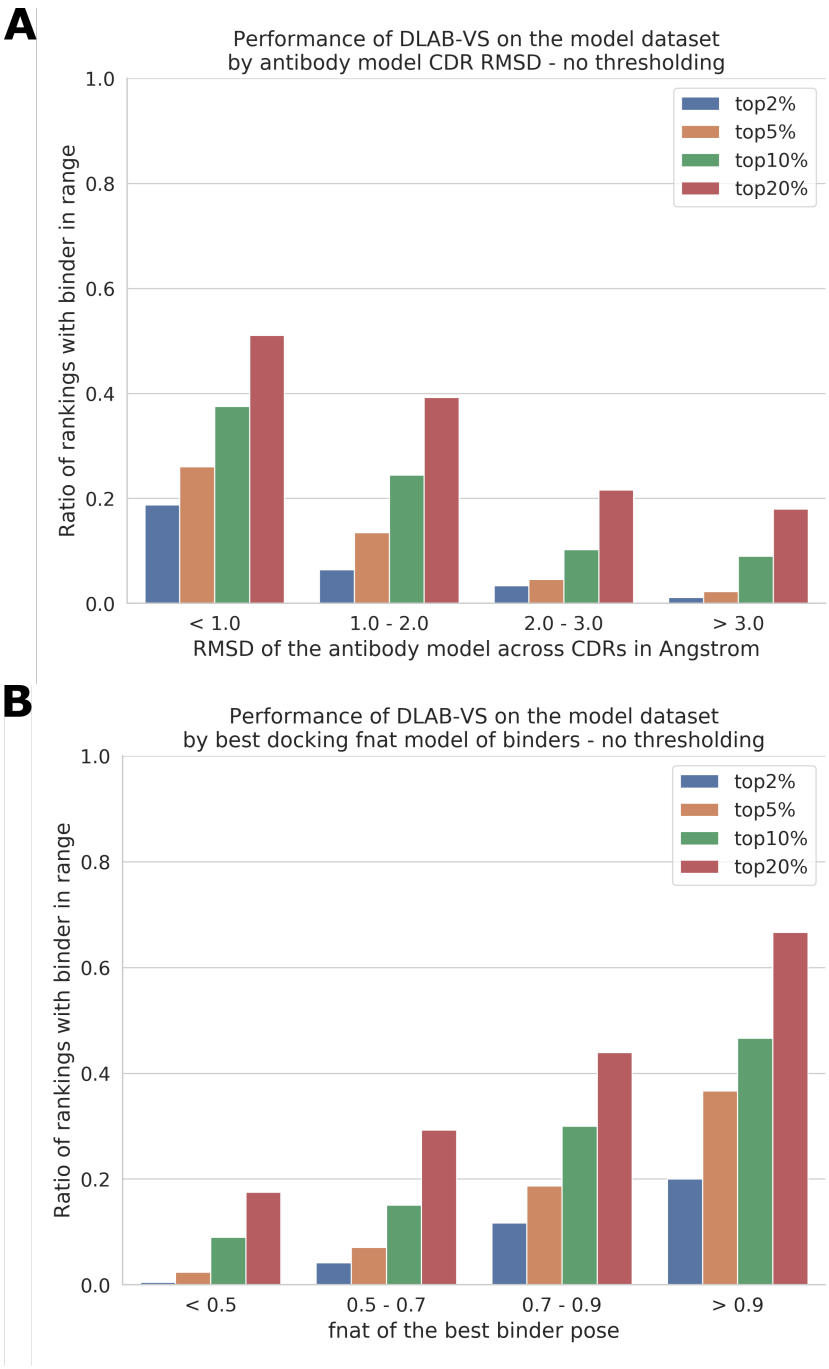

Figure 4.10: Dependence of the performance of DLAB-VS+ZDOCK on antibody model quality (A), measured via RMSD of the CDR regions to the corresponding crystal structure, and docking quality (B), measured via the highest fnat achieved in the top 500 docking poses generated by ZDOCK. Both good antibody models and high-quality docking poses correlate with high classification performance.

High-quality antibody models can be classified with high accuracy. Only considering antigen targets for which the cognate antibody was modelled with CDR RMSD < 1.0 Å, the binding antibody is ranked by DLAB-VS+ZDOCK in the top 2% and top 10% of antibodies for 18.8% and 37.5% of antigen targets respectively. Similarly, on targets for which the best docked pose of the cognate antibody amongst the top 500 poses generated by ZDOCK has *fnat* > 0.9, the binding antibody is ranked by DLAB-VS+ZDOCK in the top 2% and top 10% of antibodies for 20.0% and 46.7% of antigen targets respectively.

To further ensure that the increase in performance with lower CDR RMSD is not solely due to better antibody models being docked more accurately (as demonstrated in the previous chapter), I implemented a docking free dataset generation strategy, in which modelled antibodies are superimposed on the crystal structure of the binding antibody in complex with the cognate antigen and relaxed in the Rosetta force field as described in section 4.3.3.2. Training on this dataset and, as above, stratifying the test set by model RMSD, confirms that model quality strongly drives the ability of DLAB-VS to accurately distinguish between binders and non-binders (see Fig. 4.11). It should be noted that this pose generation strategy would not be suitable for a realistic virtual screening use case, as it requires knowledge of the correct orientation of the cognate antibody in order to calculate the starting orientation.

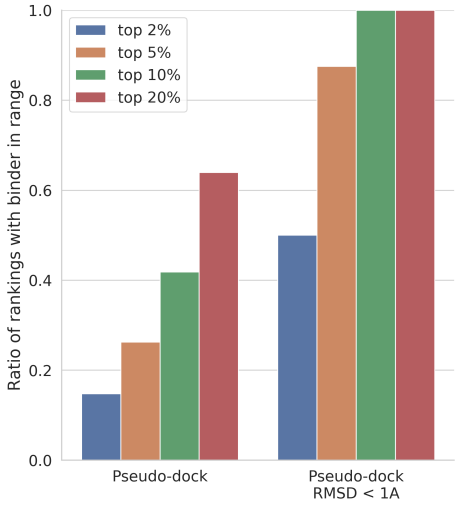

Figure 4.11: Performance of DLAB-VS on the align-superimpose model data set. High model quality correlates with strongly increased classification performance.

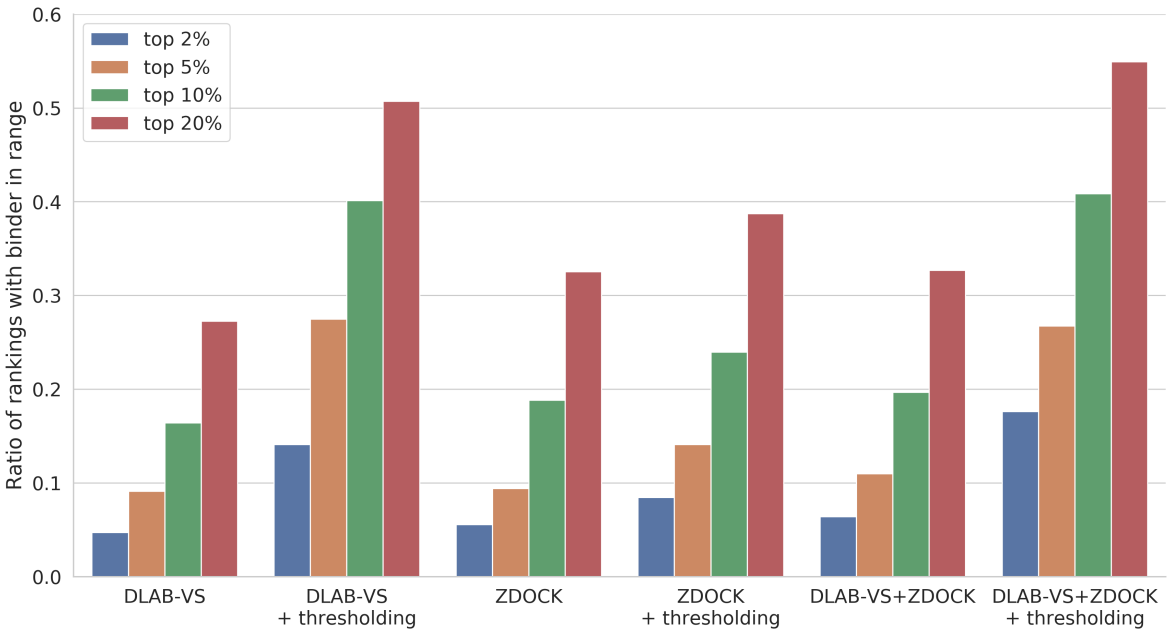

Figure 4.12: Influence of DLAB-Re thresholding on binder classification performance. Using the thresholding strategy detailed in section 4.3.3.4 to discard 80% of antigen targets leads to a large improvement in performance of both the DLAB-VS and the DLAB-VS+ZDOCK model and to a smaller improvement of the classification performance of ZDOCK scores.

4.4.9 Using DLAB-Re-derived data enables identification of well ranked targets

As demonstrated in the previous section, the performance of the DLAB-VS+ZDOCK model is highly dependent on the quality of the docking poses for which the score is calculated as well as the quality of the antibody model. As described in the previous chapter, the CNN-based rescoring model DLAB-Re enables the selection of well-docked antibody-antigen pairings. Therefore, to further improve the performance of the DLAB-VS+ZDOCK classifier, I used the output from DLAB-Re to identify antibody models with a high likelihood of being well docked from the model data set. To test if this would improve results, for each of the training cross-validation folds, I only considered antigen targets for which the DLAB-Re-max score of the cognate antibody was within the top 20% of DLAB-Re-max scores of antibody-antigen pairings within that fold. On these targets, where the cognate antibody was predicted to be well docked by DLAB-Re, the cognate antibody was ranked in the top 2% for 17.6% of antigens and in the top 10% for 40.8% of antigens. This improvement was reliant on using the combined DLAB-VS+ZDOCK model, using the same approach while ranking by the ZDOCK output scores alone only increased the classification performance marginally (see Fig. 4.12).

4.4.10 DLAB-VS+ZDOCK performance generalises to a held-out test set

The DLAB pipeline for antibody virtual screening developed through the experiments described in this chapter, displayed in Fig. 4.13 and contextualised in Fig. 1.12 consists of modelling a set of antibody structures from sequence using ABodyBuilder, docking those antibody

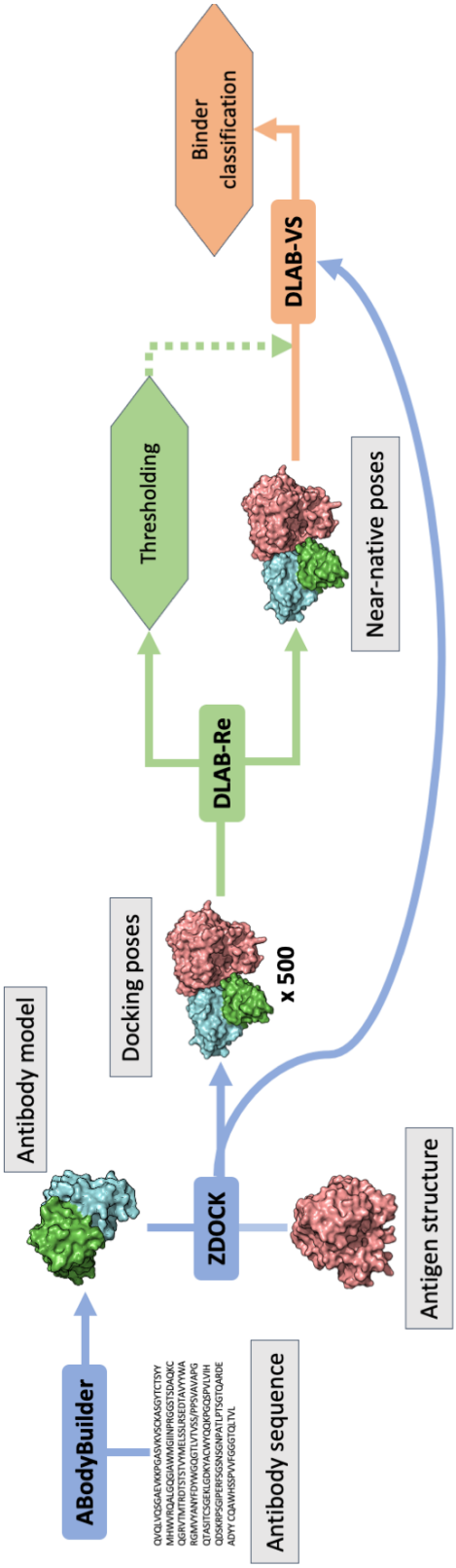

Figure 4.13: Flowchart depicting the DLAB pipeline. Blue boxes/arrows indicate external tools used to generate models and docking poses, green boxes/arrows indicate where DLAB-Re fits in the pipeline and similarly, orange boxes/arrows indicate DLAB-VS. The blue arrow connecting ZDOCK and DLAB-VS indicates the use of ZDOCK scores in the binder classification model.

models against target epitopes using ZDOCK, reranking, assessing and thresholding the docked poses using DLAB-Re and classifying the antibody antigen binding using a combination of the DLAB-VS ensemble model and the ZDOCK output.

In order to test the performance of the DLAB pipeline on a completely unseen test set, I ran the pipeline on the post-snapshot model data set (see section 4.3.6.1). On this set, an ensemble of all 40 DLAB-VS models trained during the clustered cross-validation training on the model data set and the ZDOCK output scores were used to rank the cognate antibody model as well as 50 non-cognate antibody models per target antigen. The ensemble achieved higher performance on this test set than on the previous test cases (both with and without the DLAB-Re selection criterion).

In order to calculate how different the post-snapshot model data set was from the training set and the potential for this to influence model performance, I clustered the antibody CDR sequences in both the model data set and the post-snapshot model data set at 90% identity. Of the new CDR sequences, 17.3% clustered with at least one CDR sequence in the snapshot. On the cognate antigen targets for those antibodies, the large ensemble performed exceptionally well both before and after DLAB-Re thresholding (binder in top 2% for 18% and 57% of antigen targets respectively, see Fig. 4.14). On the subset of new additions to SAbDab without overlap to the snapshot, the performance using the 40-model DLAB-VS+ZDOCK ensemble is similar to the performance on the snapshot using 10 folds of 4-model ensembles (see Fig. 4.14). The generalisation performance of DLAB-VS is therefore not based on overlap between the training set and the post-snapshot test set. I further demonstrated this by assessing the performance

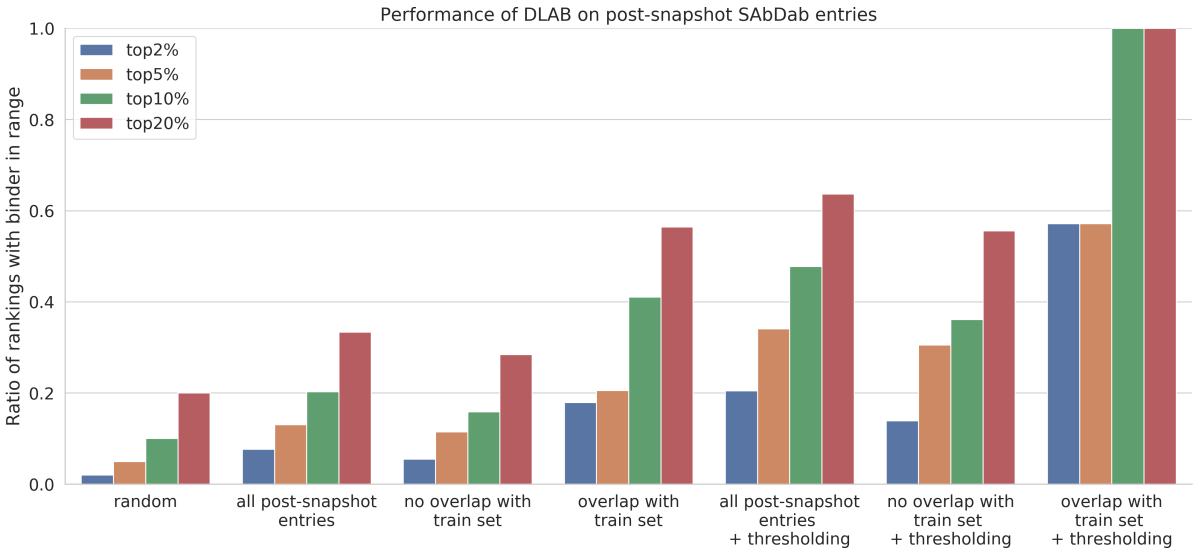

Figure 4.14: Overview of the performance of DLAB-VS+ZDOCK on the post-snapshot model data set. I compare the random expectation value ("random") to the whole post-snapshot model set ("all post-snapshot entries"), the antigen targets within the set for which the binding antibody does ("overlap with train set") or does not ("no overlap with train set") cluster with at least one antibody from the model data set at 90% CDR sequence identity. For each of these three options, the improved performance upon using the DLAB-Re-max score to discard 80% of antigen targets as described above is shown as well ("+ thresholding").

of the DLAB pipeline on the set of post-snapshot targets with cognate antibodies with at most 85%, 80%, 75% or 70% CDR sequence identity to any antibody in the training set. I observed no significant change in performance between the different thresholds, demonstrating that the predictive performance of the DLAB pipeline is generalisable (see Fig. 4.15).

4.4.11 DLAB-VS+ZDOCK performs well on realistic use case test sets

Lastly, I generated several test cases which are realistic representations of potential use cases of the DLAB pipeline or similar virtual screening pipelines.

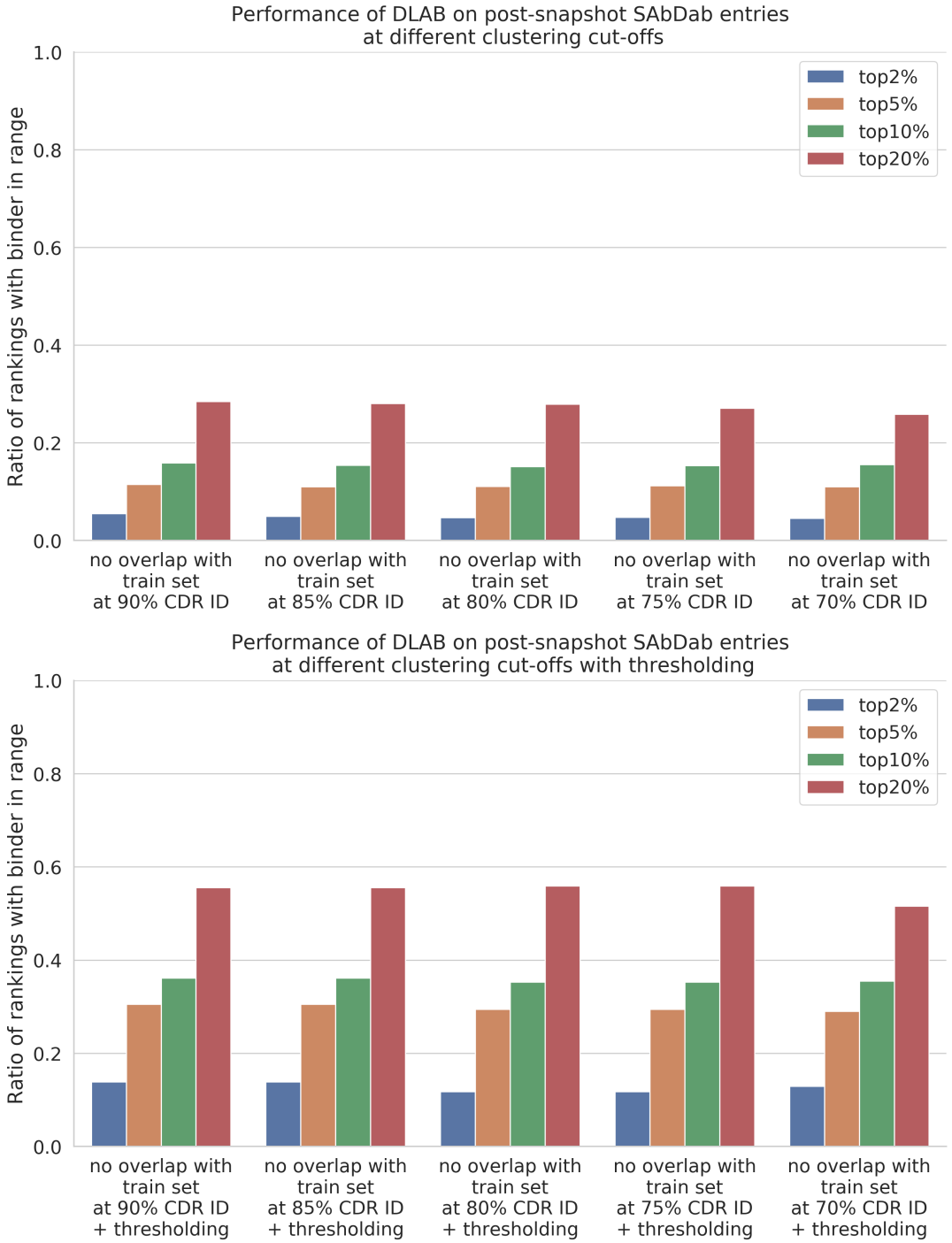

Figure 4.15: Performance of DLAB-VS+ZDOCK on the post-snapshot dataset at different percentage CDR sequence identity cut-offs for overlap with the training...

Caption continues on next page.

Figure 4.15: continued from previous page
...set without **(Top)** and with **(Bottom)** DLAB-Re thresholding. The performance of the DLAB-VS+ZDOCK model only marginally declines when limiting the allowed overlap to training set antibodies from 90% CDR ID to 70% CDR sequence identity, demonstrating that the model performance is generalisable.

In a real use case for a virtual screening pipeline like DLAB, as laid out in section 1.4.4 and Fig. 1.12, an antibody model library (AML) consisting of modelled antibody structures efficiently covering a large part of antibody sequence space would be screened against a target epitope of interest and likely contain more than one binder against a target epitope (Raybould et al. 2021b). To simulate this case, I created two additional test sets.

I clustered the epitopes in the model data set as described in section 4.3.6.2 and created two clusters with more than ten antibodies in each cluster, binding against lysozyme and HIV GP-120 respectively (see Fig. 4.16). For each cluster, I combined the binding and non-binding antibody-antigen pairings for the target epitopes and assessed the ability of DLAB-VS+ZDOCK to promote the binding antibodies to the top of the two rankings, both with and without the DLAB-Re thresholding approach described above. On average, 11.4% of the binding antibodies were found in the top 2% of the ranking before DLAB-Re thresholding, with 52% of binding antibodies found in the top 20% of the ranking. Using the DLAB-Re thresholding approach to discard 80% of pairings, this ratio increased to 12.8% and 70.8% respectively (see Fig. 4.17).

I further generated a second test case simulating this scenario using antibodies binding to the receptor binding domain on the SARS-CoV-1 or SARS-CoV-2 spike protein (see section 4.3.6.3).

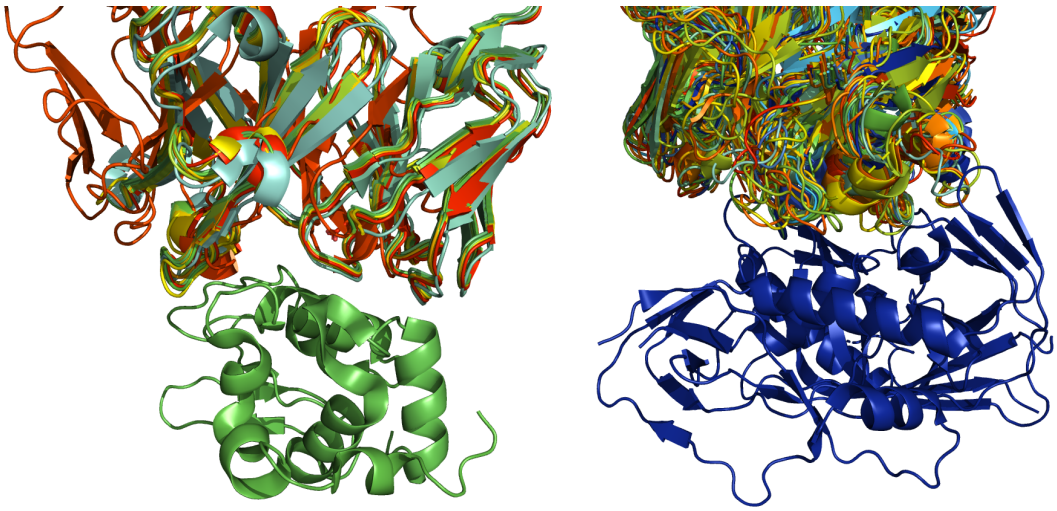

Figure 4.16: Visualisation of the two epitope clusters used for the epitope cluster test set. **(Left)** The lysozyme epitope cluster. The antigen is coloured green. **(Right)** The HIV GP-120 epitope cluster. The antigen is coloured blue.

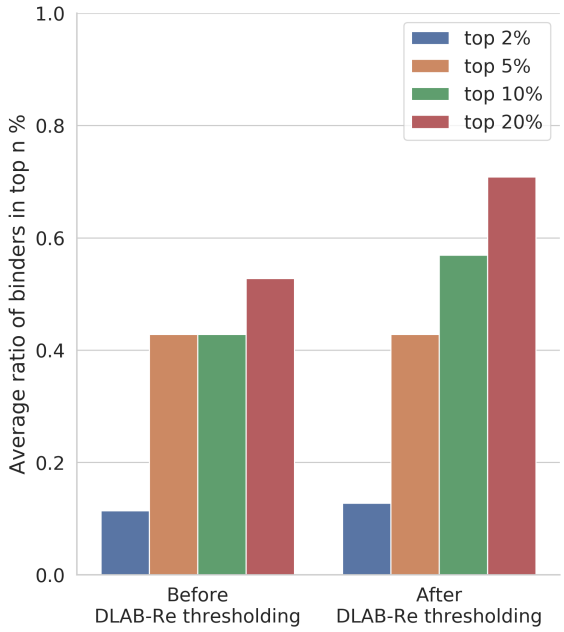

Figure 4.17: Performance of DLAB-VS+ZDOCK on the epitope cluster test set. Other than previous bar plots in this chapter, the y-axis shows the ratio of the positive binders against the same epitope which were ranked in the top 2%, top 5%, top 10% or top 20% of the overall ranking.

Using this test set, containing confirmed binding antibodies to SARS-CoV-1 and clonotype matches to those binders in antibody repertoires of SARS-CoV-2 patients, I investigated the ability of the DLAB pipeline to retrieve the binding antibodies from as set of randomly chosen antibodies from an OAS derived AML as described in section 4.3.6.3.

The DLAB pipeline achieved high performance, ranking 13 of the 18 binding antibodies (72%) in the top 10% for the SARS-CoV-1 RBD and 5 of the 18 (28%) for the SARS-CoV-2 RBD (see Fig. 4.18 Top). This performance declines when evaluating ranking performance against a H3-length-matched subset of the AML (see Fig. 4.18 Bottom), with only 8 of the 18 (44%) binding antibodies in the top 10% for the SARS-CoV-1 RBD and 1 of the 18 (6%) for the SARS-CoV-2 RBD. Ranking against a H3-length-matched AML subset is likely harder due to the higher degree of shape similarity between antibody CDRs with matching length.

Another use case of interest is determining whether mutations in the antigen can disrupt antibody binding. In order to test whether this task is accessible to structure-based deep learning tools, I created a data set of antibodies confirmed to bind against the SARS-CoV-2 wild-type receptor binding domain (RBD) while also being confirmed not to bind against at least one SARS-CoV-2 RBD variant (see section 4.3.6.4 and Table B.4). I ran the data set through the DLAB pipeline as described in Fig. 4.13 and assessed whether the DLAB-VS+ZDOCK output score consistently scored the antibody-wild-type pair higher than the antibody-variant pair. For the 14 antibody-variant pairs in the data set, the DLAB-VS+ZDOCK score of the antibody-wild-type pair was higher than the score of the antibody-variant pair in 13/14 cases. This result indicates that the variant classification problem is accessible to structure-based deep learning

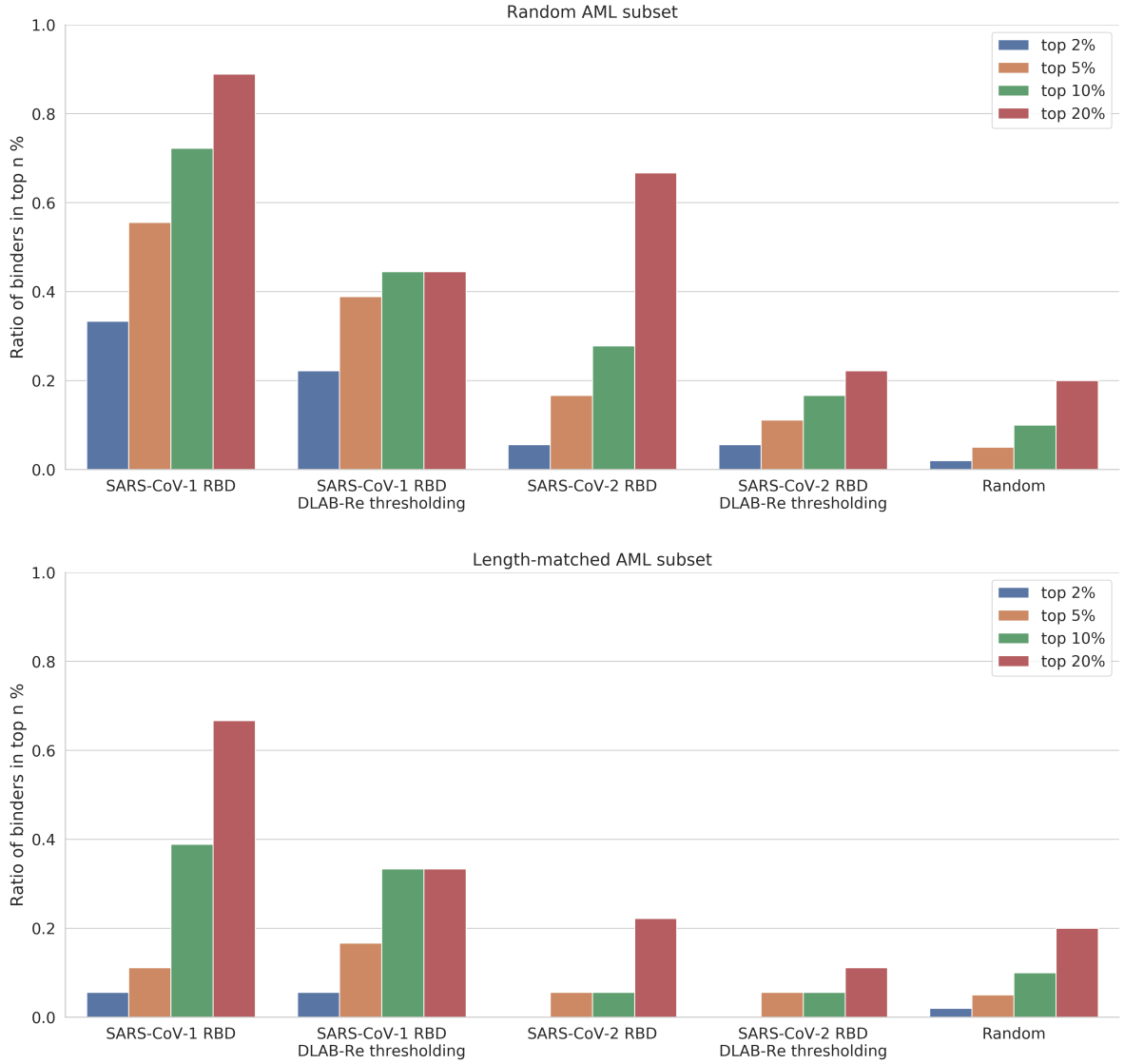

Figure 4.18: Performance of DLAB-VS+ZDOCK on the SARS-CoV-1/2 test set. **(Top)** Ranking performance against the random AML subset. **(Bottom)** Ranking performance against the CDR H3-length-matched AML subset. For both graphs, the y-axis shows the ratio of confirmed SARS-CoV-1 binders or clonotype matches against confirmed binders ranked in the top 2%, top 5%, top 10% or top 20% of the ranking for either the SARS-CoV-1 or SARS-CoV-2 epitope.

tools, such as DLAB-VS.

4.5 Discussion

One of the major shortcomings of current computational antibody drug discovery methods is the lack of structure-based, early-pipeline screening tools to identify promising candidate antibodies.

In this chapter, I have shown that my virtual screening tool, DLAB, can identify putative binders to a given epitope in several different settings. The complete DLAB pipeline of docking followed by DLAB-Re and DLAB-VS enriched binders both against the background of non-binding SAbDab-deposited sequences as well as in a more realistic usage scenario against antibody sequences drawn from antibody repertoire data.

On the crystal structure data set with highly accurate antibody structures and docking poses, both DLAB-VS and ZDOCK are able to strongly enrich binders. In the case of model antibodies docked to antigens, where both model and docking quality have to be considered, ZDOCK and DLAB-VS approaches fail to achieve strong discrimination between binders and non-binders. These results are in line with previously published work on the ability to classify cognate antibodies through cross-docking analysis (Kilambi and Gray 2017). However, incorporating insights from small molecule virtual screening studies into the training set composition and employing ensembles of CNN models as well as combining the DLAB-VS and ZDOCK scores improved performance. The combination of several CNN models as well as the combination of DLAB-VS and ZDOCK output are bagging approaches, improving the overall

model performance due to the reduction in model variance resulting from the combination of models with (partially) uncorrelated errors (Maclin and Opitz 1999).

On a realistic use case, using the SARS-CoV-1 and SARS-CoV-2 receptor binding domain epitope as the target epitopes, I have demonstrated the utility of the DLAB pipeline, enriching binding antibodies against a set of 300 decoys drawn from an OAS derived AML. On a second test set using the SARS-CoV-2 receptor binding domain epitope, DLAB-VS+ZDOCK correctly scored antibody escape variants lower than the cognate epitopes for 13 of 14 antibody-variant pairs.

The DLAB pipeline has been trained specifically on a combination of ABodyBuilder and ZDOCK. The use of a different input pipeline would likely require additional fine-tuning of the weights of both DLAB-Re and DLAB-VS. As mentioned in chapter 3, one natural extension would be the use of flexible docking approaches, which could improve the input docking poses but would be computationally expensive given the scale of experiments needed in a high-throughput setting. Another possible extension, which is discussed in more detail in chapter 6, is the application of transfer learning approaches to the DLAB pipeline, which could enable learning protein-protein interaction features from the much larger set of known protein-protein complex structures: the Database of Interacting Protein Structures (Townshend et al. 2019) contains over 42,000 complex structures, more than 40 times the amount of data used to train the models detailed in this chapter.

In this chapter, I have demonstrated the applicability of structure-based deep learning approaches to the antibody virtual screening task. Methods such as DLAB will improve with

increasing availability of structural antibody data as well as improved antibody modelling and improved fast docking methods. DLAB demonstrates the potential of structure-based deep learning approaches to supplement traditional experimental screening approaches and sets a course for structure-based virtual screening methods for antibody drug discovery.

The DLAB source code and trained models are available at github.com/oxpig/dlab-public.

5 | Antibody-antigen binding classification with equivariant graph neural networks

Contents

5.1 Chapter motivation

Recent advances in machine learning for protein structure generation and analysis, including the development of AlphaFold2 (Jumper et al. 2021), have highlighted the potential of equiv-

ariant graph neural networks. In this chapter I evaluate the ability of several rotation and translation equivariant graph-based machine learning approaches to learn from antibody-antigen interaction data and I detail the development of the graph-based classification model DLAB-EG. I find that this change in model alone leads to a significant improvement over the DLAB-VS model detailed in the previous chapter.

5.2 Introduction

In recent years, graph neural networks (e.g. Eismann et al. (2020a), Jing et al. (2020), Abanades et al. (2021), and Pittala and Bailey-Kellogg (2020)) and geometric deep learning approaches (e.g. Dai and Bailey-Kellogg (2021) and Gainza et al. (2020)) operating on graphs or point-clouds of atom- or residue-level input data have emerged as a promising class of machine learning approaches for protein structure and activity prediction. At the same time, the inclusion of roto-translational equivariance in model architectures has become a common feature amongst well performing neural network models operating on molecular data (e.g. Weiler et al. (2018), Fuchs et al. (2020a), Hutchinson et al. (2021), and Satorras et al. (2021)). This development is exemplified by the results recently achieved in structure prediction by methods employing these features, particularly by the AlphaFold2 model (Jumper et al. 2021), discussed in more detail in section 1.2.2.1. Recent studies have also achieved state-of-the art performance on other prediction tasks on protein data using machine learning models characterised by these features, such as model quality assessment (e.g. Jing et al. (2020)) and complex pose assessment (e.g. Eismann et al. (2020a)).

These graph-based models offer several advantages over the grid-based convolutional model architectures used in chapters 3 and 4. They are more compact, requiring less trainable parameters to achieve good performance while also often requiring less memory. The memory required to store grids for input into CNN models grows cubically with the size of the input grid, while graphs of similar data grow quadratically with the number of nodes in fully connected graphs and linearly in graphs with local connectivity. Graphs further remove the necessity to consider voxel occupancy by atoms/residues. For grids, it may be sensible to have the same atom occupy several adjacent voxels (Ragoza et al. 2017) to accurately encode biological data and reduce sparsity, graphs on the other hand encode this information implicitly in node features and connectivity. Lastly, equivariance or invariance to rotation and translation of input data reduces the need for data augmentation at train time. Except for specialised CNN architectures (Weiler et al. 2018), CNNs are equivariant to translation, but not to rotation and therefore often require data augmentation through random rotation of the input data (e.g. Schneider et al. (2021b)). These differences make graph neural networks well suited for working on protein data including antibody-antigen structural data.

Over the last two years, several open-source implementations of equivariant graph neural networks have been published (e.g. Fuchs et al. (2020a), Hutchinson et al. (2021), Satorras et al. (2021), Jing et al. (2020), and Eismann et al. (2020a)). In this chapter, I am using the model architectures proposed by Satorras et al. (2021) and Fuchs et al. (2020a) alongside the non-equivariant architecture proposed by Kipf and Welling (2017) to categorise and benchmark the potential of these methods on the antibody domain. I trained these graph neural network model types on the same antibody-antigen binding tasks as DLAB-VS to develop DLAB-EG,

an equivariant graph neural network model for antibody virtual screening. In the following, I show that using graph-based neural networks enables the training of fast, efficient, and well performing binder classification models and presents a promising direction for future research.

5.3 Methods

5.3.1 Training set compositions

In order to train and test the graph neural networks described in this chapter, I used training and test sets derived from the same structures and using the same training/validate/test splits as in the previous chapter. While there was additional structural data available at the time of training these networks, keeping the data sets the same meant I could directly compare the performance of the new graph neural network models to the previously trained CNN model DLAB-VS.

For the training of graph neural network (GNN) models to perform classification on the crystal structure data set, I used the same training set composition as in the previous chapter, including the top ranked docked pose for each antibody-antigen pairing.

For the training of GNN models to perform classification on modelled antibodies, in addition to the training set compositions labelled as training sets 4 and 5 in chapter 4 (see sections 4.3.5.3 and 4.4.3 and Table 4.1), referred to here as training sets g-1 and g-2, I also explored the impact of exposing the model to antibody crystal structures at train time, but not at test time. To this end, I created a training set (g-3) solely composed of the top ranked pose from the crystal structure data set for both antibody-antigen pairings labelled binding and non-binding.

ID	crystals structures	modelled structures
g-1	no	DLAB-VS training set with *fnat* cut-off 0.7
g-2	no	DLAB-VS training set with *fnat* cut-off 0.5
g-3	yes	no
g-4	yes	DLAB-VS training set with *fnat* cut-off 0.7
tp-a	no	top10
tp-b	top10	top10

Table 5.1: Training sets for graph model training. g-1 to g-4 are the training sets used for training of the single-pose graph neural networks. "DLAB-VS training set with *fnat* cut-off 0.7/0.5" refers to the training set compositions labelled training set 4 and 5 in the previous chapter. tp-a and tp-b are the training sets used to train the multi-pose graph neural networks.

Secondly, I combined training sets g-1 and g-3 to include the docked poses of both antibody crystal structures and antibody models with the previously described *fnat* cut-offs applied. This training set is in the following referred to as g-4.

For the training of GNN models that derive a classification score from ten poses of the same antibody-antigen pairing (referred to in the following as ten-pose models and described in section 5.3.5.5), I created two data sets. Data set tp-a contains the ten top ranked poses for each modelled antibody-antigen pairing, regardless of pose *fnat*. Data set tp-b contains the data in data set tp-a as well as the ten top ranked poses for each crystal structure antibody-antigen pairing.

All training sets used in this chapter are summarised in Table 5.1.

5.3.2 Test sets

In order to accurately compare the previously trained CNN models to the GNN models trained in this chapter, I used the same test sets as in the previous chapter: the held-out test set of antibody structures deposited after the creation of the SAbDab snapshot (see section 4.3.6.1), the epitope cluster test case (see section 4.3.6.2) and the SARS-CoV-1/2 test case (see section 4.3.6.3) as well as the SARS-CoV-2 variant binding test case (see section 4.3.6.4). I evaluated the performance on these test cases equivalently to the previous chapter, only evaluating the GNN model with the best performance during cross-validation on these test cases to avoid multiple comparisons.

5.3.3 Generation of docked poses

As in previous chapters, I generated antibody-antigen poses using ZDOCK, following the same procedures described in detail in section 3.3.4.

In the previous chapters, I generated each docked pose prior to model training. This resulted in large storage requirements, as up to 500 poses were generated per antibody-antigen pairing: the model structure data set required roughly one terabyte of hard-drive space prior to training. To alleviate this bottleneck, for the graph generation in this chapter, I generated atom coordinates for the docked poses on the fly during graph generation based on the ZDOCK output, significantly reducing the storage required (all data sets used in this chapter occupied a combined 14 gigabytes of hard drive space, a reduction by two orders of magnitude).

5.3.4 Graph generation

The neural networks evaluated in this chapter use graphs as their input. Therefore, I evaluated several approaches to convert docked poses of antibody-antigen pairings into graphs. While the models tested differed in the specifics of node and edge annotation within the graph (explained in section 5.3.5), the overall graph generation process was shared across models and is schematically shown in Figure 5.1. After generation of node and edge data, the graphs themselves were generated using the `pytorch-geometric` graph data class (Fey and Lenssen 2019).

5.3.4.1 Graph generation based on distance to interaction center

Initially, I implemented a graph generation method mirroring the grid generation for DLAB-VS, referred to in the following as the DLAB-input graph generation. The interaction center point between antibody and antigen was identified as described in section 3.3.8 and all atoms within 24 Å of the center point were included in the graph as nodes. Edges were created between all atoms within 4 Å of each other and labelled as either ”inter" (between antibody and antigen) or ”intra" (within one molecule) edges.

Graphs generated using this method were large, containing up to 10,000 nodes and 100,000 edges. This led to slow model training as large graphs limited the size of each minibatch passed through the network due to the available GPU memory.

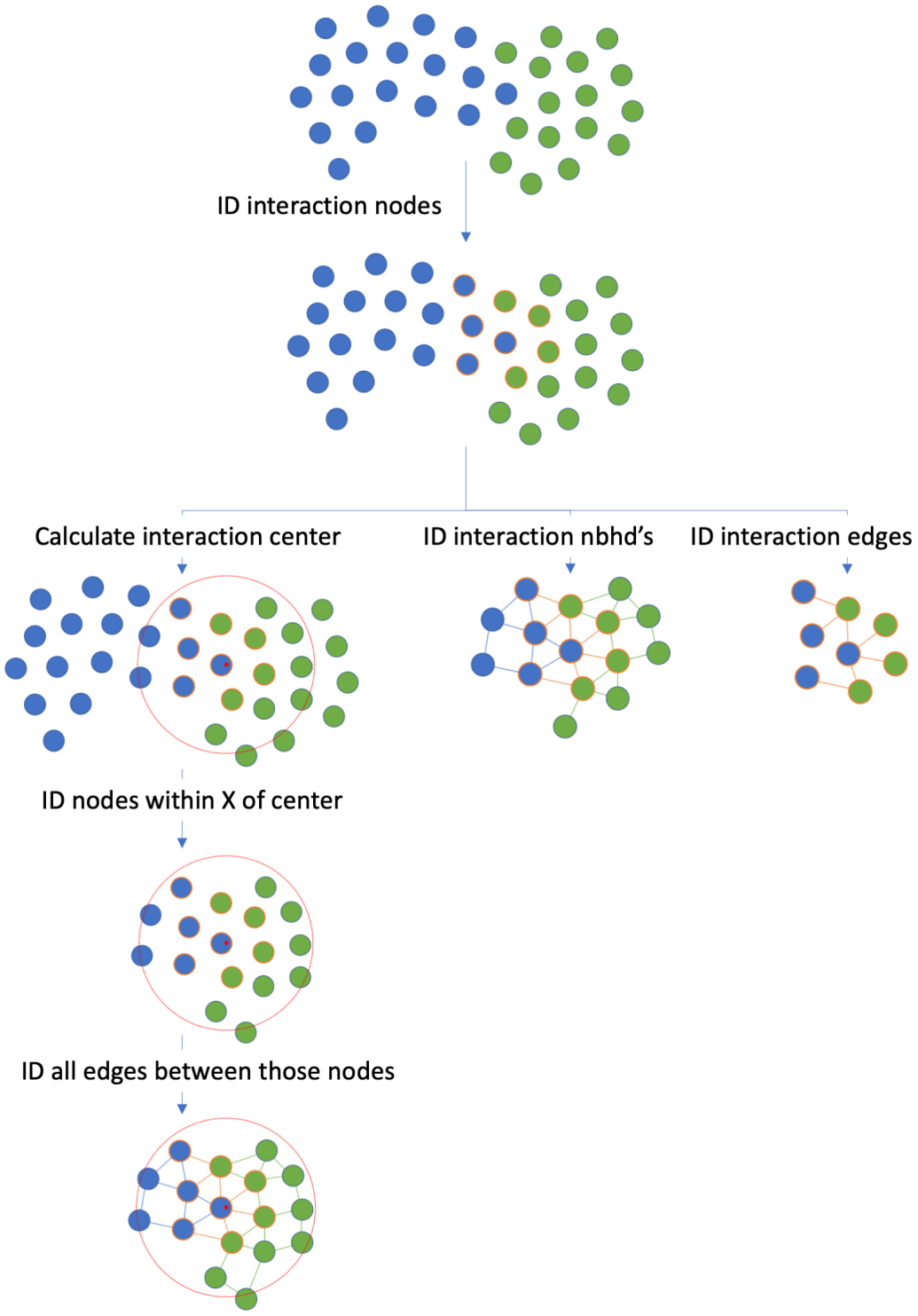

Figure 5.1: Caption on next page.

Figure 5.1: continued from previous page

Schematic overview of the graph generation algorithms used for the GNN models. A docked pose between two interaction partners is comprised of a set of nodes (atoms or residues), depicted as blue and green circles respectively. Interacting nodes are identified based on distance to the interaction partner (orange outlined nodes). Depending on the graph generation scheme, the graph is generated by (**Left)** identifying all nodes within a predefined radius from the interaction center (marked in red) and adding edges between them based on a $4\,\text{Å}$ distance cut-off, (**Middle**) identifying all nodes within $4\,\text{Å}$ of an interaction node and adding edges between those based on the same interaction cut-off or (**Right**) only using the interaction nodes and the edges connecting them to the interaction partner. Edges between the two interaction partners are coloured orange, edges within the interaction partners are coloured blue or green respectively.

5.3.4.2 Interface graphs

Unlike the grids for input into CNNs like DLAB-VS, graphs are not required to have a particular shape or size in order to be accessible to a particular GNN architecture. In order to reduce the size of the graphs generated and to evaluate different methods of graph generation, I therefore implemented two other approaches which did not generate the graphs based on a fixed radius around the interaction center.

Interface nodes were identified based on a distance cut-off ($4\,\text{Å}$ for atom nodes, $10\,\text{Å}$ for residue Cα nodes). These pairwise distance calculations were accelerated for atom nodes by first calculating Cα distances, and only calculating all-heavy atom distances for residues with Cα atoms within $15\,\text{Å}$ of an interaction-residue Cα atom. Graphs were created from the interface nodes using two approaches (see Fig. 5.1).

Interface neighbourhoods For this approach, all nodes within the distance cut-off of an interface node were included in the graph, with edges created as in the DLAB-input graph generation method.

Interface edges For this approach, only the interface nodes were included in the graph and the only edges used were those between the interaction partners.

5.3.4.3 Node types

I used two node types for the graphs generated in this chapter. Unless otherwise specified, nodes are protein heavy atoms. In this case, the interaction distance cut-off used was $4\,\text{Å}$ and each node was parameterised using the `libmolgrid` typing scheme described in section 3.3.8. This node typing scheme is referred to as the "atom" scheme in this chapter. During training of the EGNN models and the ten-pose EGNN models, I further implemented a scheme in which only Cα atoms were used as nodes in the graph and each node was labelled with the corresponding residue type. For this node definition, I used a $10\,\text{Å}$ interaction cut-off. In the following, this node typing scheme is referred to as the "residue" scheme.

5.3.5 Model architectures

I assessed the potential of several GNN architectures for antibody-antigen virtual screening. While novel GNN architectures are not regularly evaluated on protein structures, many architectures are evaluated and compared on small molecule property prediction on the QM9 dataset (Ramakrishnan et al. 2014). Where appropriate, I therefore based my initial implementation of

the models described here on the architecture evaluated on the QM9 dataset.

5.3.5.1 Graph convolutional network

I implemented a graph model based on graph convolutional network (GCN) layers (Kipf and Welling 2017). This model served as a baseline for graph neural network performance on the antibody-antigen binding classification task, as it included graph convolutions but lacked some of the features commonly used in state-of-the-art graph models for structural biological data: edge features, roto-translational equivariance and attention. The architecture is shown in Figure 5.2 and detailed in Appendix table C.1 and was implemented using `pytorch-geometric`.

5.3.5.2 SE(3)-Transformer

In order to evaluate state-of-the-art equivariant GNNs, I adapted the SE(3)-Transformer model (Fuchs et al. 2020a) to the antibody-binding classification task. The model was implemented using the code supplied in the author's GitHub repository (Fuchs et al. 2020b). This model, which has been used in several protein machine learning tools over the last year (e.g. Baek et al. (2021)), uses a series of residual blocks with attention as described in section 1.5.2. The architecture of the SE(3)-Transformer used in this thesis is shown in Fig. 5.3 and detailed in Appendix table C.2. Other than the GCN and the EGNN architecture, this model was implemented using the DGL python package (Wang et al. 2019a), as the author's implementation was based on DGL.

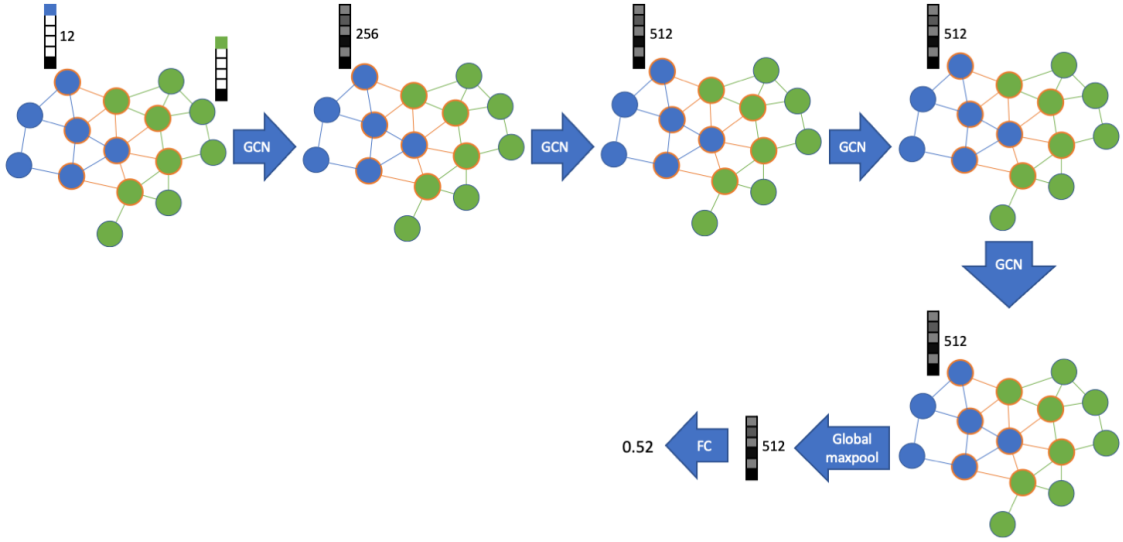

Figure 5.2: Schematic overview of the GCN architecture. Nodes are labelled with a one-hot vector encoding node-type and an additional bit encoding membership of the node to the antibody or antigen sub-graph. Four successive graph convolutional layers (GCN) are applied to the input graph, followed by max-pooling across all node vectors to derive one vector to describe the graph as a whole. This vector is then used as input to a fully connected (FC) layer to compute an output score.

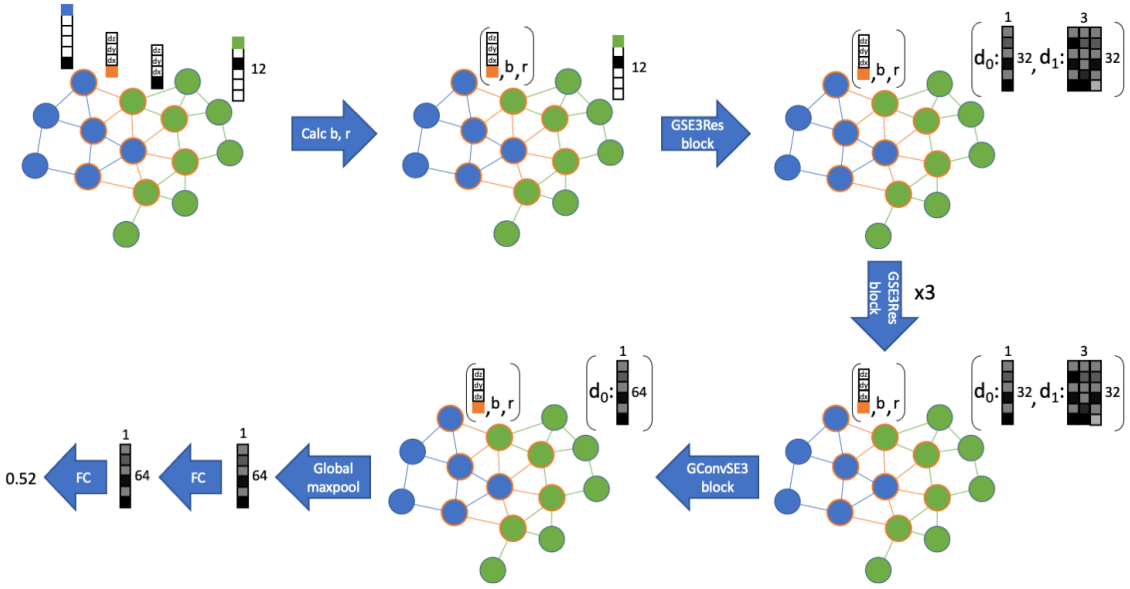

Figure 5.3: Schematic overview of the SE(3)-Transformer architecture. Nodes are labelled with a one-hot vector encoding node-type and an additional bit encoding membership of the node to the antibody or antigen sub-graph. Edges are labelled with the triplet (dx, dy, dz), signifying edge length along each axis, and an additional bit encoding whether the edge is an "intra"-edge (connecting nodes within an interaction partner) or an "inter"-edge (connecting nodes of different molecules). For each edge, first the equivariant basis kernel b and the radius r are computed (this is done once at the start of each forward pass to save computation). This information is used to perform the equivariant operations (due to computational cost limited to two degrees) in the following four GSE3Res blocks (implementing residual attention layers) and the graph convolutional block (GConvSE3). The node vectors are combined into one vector describing the graph as a whole via max-pooling and used as input to two consecutive fully connected (FC) layers to compute an output score. Throughout the computation, edge features are not changed.

5.3.5.3 EGNN

To enable the evaluation of equivariant graph neural networks without the computational drawbacks arising from the equivariant basis kernel computation required by the SE(3)-Transformer, I implemented several architectures based on the E(n)-equivariant graph neural network (EGNN) published by Satorras et al. (2021), which has recently been employed successfully on antibody structural data (Abanades et al. 2021) as well as general protein data (Ganea et al. 2021). My model was based upon code provided by the author and implemented using `pytorch-geometric`. One of the architectures implemented is shown in Fig. 5.4. The full set of architectures tested is detailed in the results section below and Appendix table C.3.

5.3.5.4 Siamese networks for no-dock classification

To evaluate if the GNN models tested in this thesis enable antibody-antigen binding classification without docking pose generation, I implemented a docking-free classification task setup and model architecture. Antibody and antigen graphs for samples in the crystal structure dataset were defined using the interface graph generation approaches described above based on their predicted paratope and known epitope. The identification of these protein regions is described in detail in section 3.3.4.2. The graphs are rotated and translated randomly and independently in order to negate any information present in the structures. This was necessary as in the crystal structure dataset, antibody and antigen structures are extracted from the experimentally determined structure and thus while the graphs are provided to the network non-joint, their respective orientations would still allow the network to determine binding based on the binding antibodies

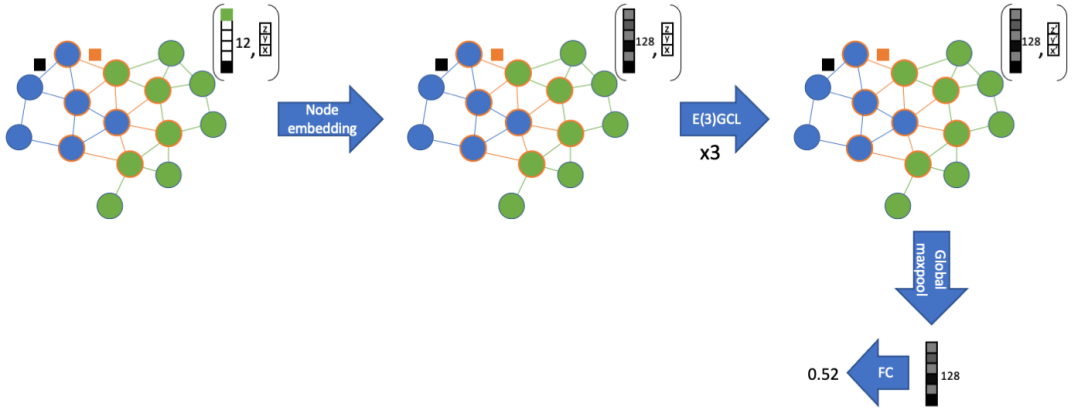

Figure 5.4: Schematic overview of the EGNN architecture. Nodes are labelled with a one-hot vector encoding node-type and an additional bit encoding membership of the node to the antibody or antigen sub-graph as well as their x, y, z coordinates. Edges are labelled with a bit encoding whether the edge is an "intra"-edge or an "inter"-edge. Node encodings are embedded into node feature vectors using a node-wise fully connected layer (node embedding). The node and edge features are used as input to the equivariant graph convolutional layers described by Satorras et al. (2021) and in section 1.5.3. The node vectors are combined into one vector describing the graph as a whole via max-pooling and used as input to a fully connected (FC) layer to compute an output score. Throughout the computation, edge features are not changed. As suggested by the author, the convolutional layers use the tanh activation function and residual skip connections in order to improve model stability during training.

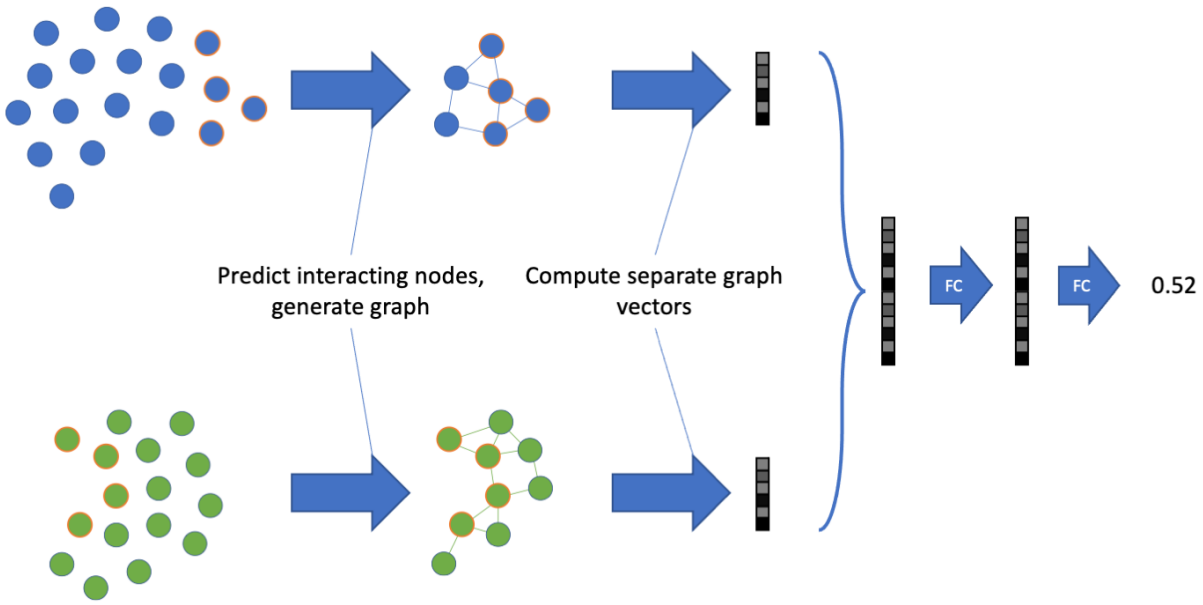

Figure 5.5: Schematic overview of the no-dock task setup. The graph representations of the interaction partners are fed into the graph neural network model separately prior to docking. Using separate networks for antibody and antigen, a vector for each molecule is generated and the vectors are concatenated to form a joint representation of the two interaction partners, which is used as input to fully connected layers to compute an output score for binder classification.

being provided in the ”correct" orientation.

To derive a classification score, the two graphs for antibody and antigen are input into a GNN model consisting of two separate GNNs used to derive a single feature vector for each antibody and antigen. These feature vectors are then concatenated, and an output score is generated using fully connected layers (see Fig. 5.5).

5.3.5.5 Multi-pose EGNN

In the previous chapter I showed that combining classification scores of several input poses can be beneficial for model performance (see section 4.4.5). I therefore implemented a model architecture which directly computes a joint classification score by passing ten interaction graphs through the same EGNN model and extracting a graph feature vector for each graph (see Fig. 5.6). The ten feature vectors are then pooled into one final vector, which is passed through a fully connected layer to arrive at a classification score. The architecture is displayed in Fig 5.6 and model parameters are detailed in Appendix table C.4.

5.3.6 Model training

Models were trained using pytorch lightning (Falcon and Cho 2020) with parameters as specified in Appendix C.1.

On the crystal data set, I trained and tested on only one train/validation/test set partition in order to quickly prototype several model types. On the model data set, I used the same ten-fold cross-validation approach during training as that used in chapter 4 for the training of DLAB-VS in order to simplify comparisons between the two model types. Models were generally trained for 20 epochs, with the exception of the SE(3)-Transformer models, for which model training was prohibitively slow[1] and which were therefore trained for four epochs only and the ten-pose EGNN model, which converged slower and was therefore trained for 40 epochs. Model weights were saved after each epoch and the model with the lowest validation loss was evaluated on the

[1] It should be noted that a considerably faster implementation of the SE(3)-Transformer is now available (Milesi 2021), but this was not yet released when I performed the experiments detailed in this chapter.

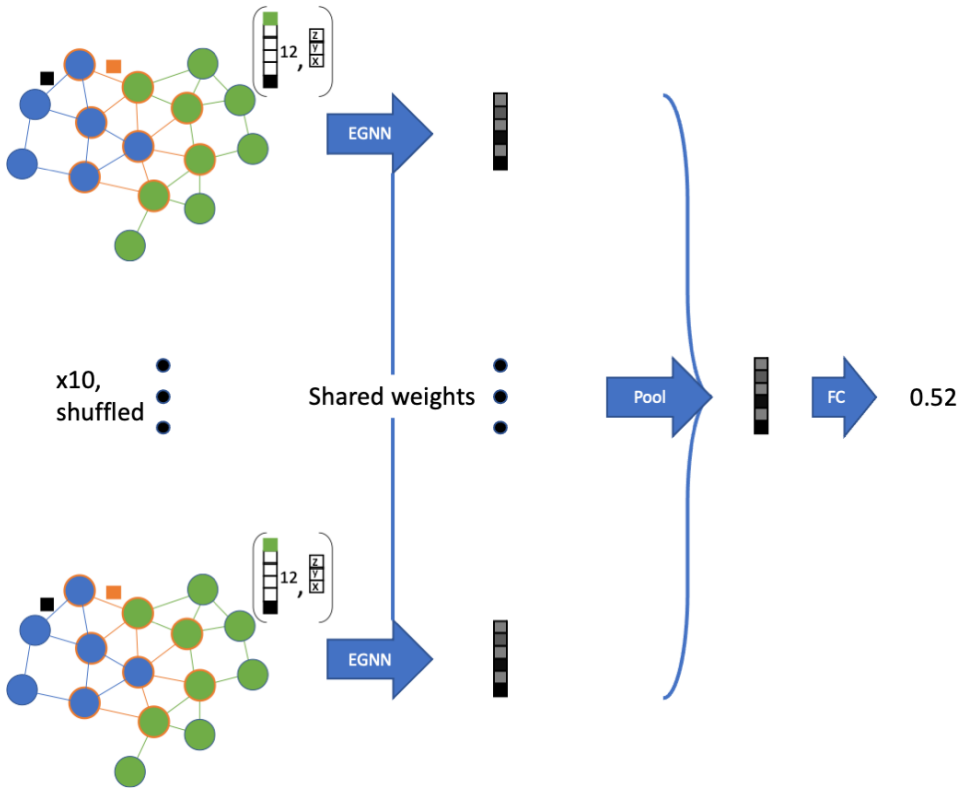

Figure 5.6: Schematic overview of the multi-pose EGNN. In each forward pass, the same EGNN model is used to generate feature vectors for graph representations of several docked poses (10 in the results presented in this thesis). The resulting feature vectors are pooled to derive a joint representation across the input poses, which is used to compute classification scores.

test set.

Due to the advantage from using model ensembles observed in chapter 4 in sections 4.4.5 and 4.4.6, when training the EGNN models on the model data set, I trained an ensemble of four models for each cross-validation fold. Further, on the model data set, I evaluated all single-pose models using the mean score predicted for the ten top ranked poses.

5.3.7 Combining GNN model scores and ZDOCK output

As described in chapter 4 (see section 4.3.5.6), I combined the output scores of the GNN models with the ZDOCK output scores via minmax-scaling and averaging.

5.4 Results

5.4.1 Graph models can improve over CNNs on the crystal structure dataset

I first evaluated if graph neural networks (GNNs) can be applied to the antibody-antigen binding classification task by training several GNN architectures and graph generation schemes on the crystal structure data set. As mentioned in section 5.3.6, for these experiments, I trained, validated, and tested the models on only one cross-validation fold in order to enable rapid implementation and prototyping of several model types.

5.4.1.1 The SE(3)-Transformer can achieve performance similar to DLAB-VS on the crystal structure data set

I first tested the SE(3)-Transformer model with the DLAB-input scheme (see section 5.3.4.1) in order to generate an initial hypothesis on the applicability of GNN models on antibody-antigen structural data. Using this input scheme, which included in the input graph atom-level nodes selected based on the same distance cut-offs used for DLAB-VS grid generation, the SE(3)-Transformer model replicated the DLAB-VS performance on the crystal structure data set: the SE(3)-Transformer model ranked the binding antibody in the top 2% and top 10% for 60.9% and 71.5% of antigen targets, similar to the performance achieved using DLAB-VS (62.6% and 72.8%, see Fig. 5.7). This performance indicated that GNNs are a valid method to use on the antibody-antigen binding prediction task.

5.4.1.2 Graph generation can be simplified for improved performance, speed, and memory requirements

The graph generation scheme used in section 5.4.1.1, while achieving results similar to DLAB-VS with the SE(3)-Transformer, suffers from the large size of the generated input graphs. The requirement to store features for all nodes and edges in the graph alongside attention weights for each edge in each graph during training meant that graph creation and transfer to GPU became computational bottlenecks. I therefore investigated the performance of two graph creation schemes which only include nodes close to the interaction site between antibody and antigens (see Fig. 5.1), leading to much smaller graphs and faster training times. The interaction neigh-

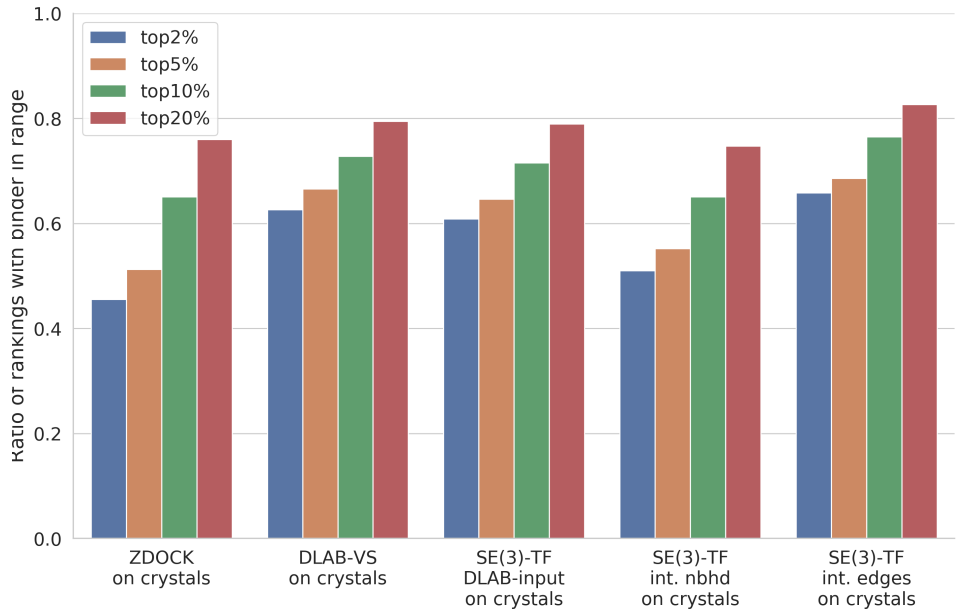

Figure 5.7: Comparison of the performance of ZDOCK, DLAB-VS, and the SE(3)-Transformer using different graph generation schemes on the crystal structure data set. The ratio of antigen targets for which the binding antibody was ranked in the top 2%, top 5%, top 10% and top 20% respectively is shown. Using the same graph generation mode mirroring grid generation for DLAB-VS (DLAB-input), SE(3)-TF performs similar to DLAB-VS. Using the interaction edges graph generation mode, SE(3)-TF performs better than DLAB-VS and ZDOCK.

bourhood graph generation scheme reduced the size of the resulting graphs by approximately 75% compared to the DLAB-input scheme, the interaction edges scheme reduced the graph size to about 3% of the DLAB-input graphs (mainly through the reduction in the number of edges in the graph).

The interaction edges graph generation scheme improved performance slightly over the DLAB-input graph generation scheme (see Fig. 5.7), ranking the binding antibody in the top 2% and top 10% for 65.81% and 76.5% of antigen targets respectively. This result did not present a significant improvement over either the SE(3)-Transformer trained with the DLAB-input graph generation scheme or the DLAB-VS CNN model, but validated the efficient interaction edges graph generation scheme, which enabled the significant improvements achieved using the EGNN model in the following sections.

In the following analysis, I used the interaction edges graph scheme for the SE(3)-Transformer and evaluated both the interaction edges and the interaction neighbourhood graph scheme for the GCN and the EGNN model.

5.4.1.3 The EGNN model enables improvement over DLAB-VS on the crystal structure data set

I next compared the performance of the SE(3)-Transformer, an attention-based equivariant GNN model with high computational cost at run time, to two simpler model architectures. The graph convolutional neural network model (GCN) (Kipf and Welling 2017) is an early graph neural network, while the E(n) equivariant convolutional neural network (EGNN) (Satorras et

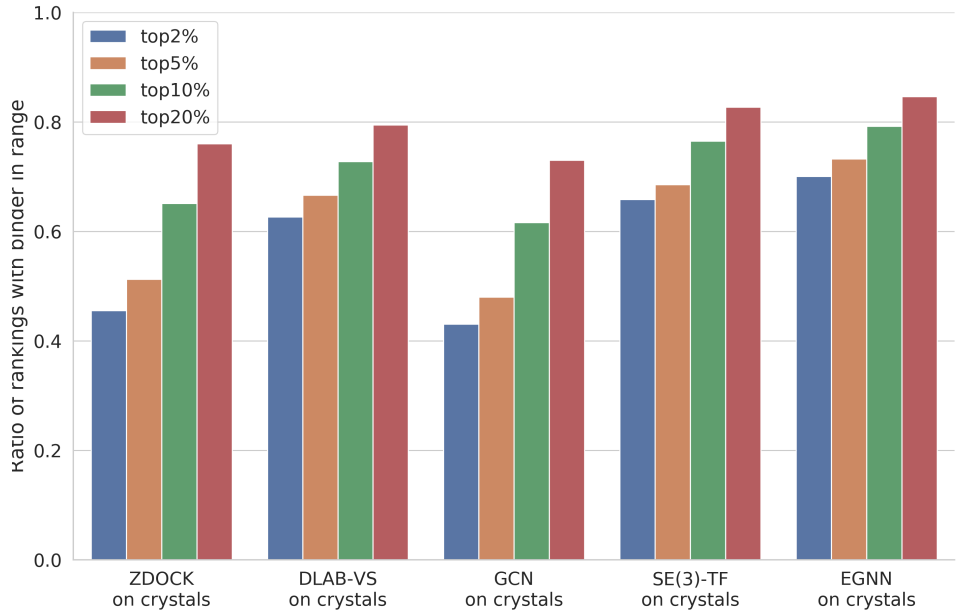

Figure 5.8: Comparison of the performance of ZDOCK, DLAB-VS, SE(3)-Transformer, GCN and EGNN on the crystal structure data set. The ratio of antigen targets for which the binding antibody was ranked in the top 2%, top 5%, top 10% and top 20% respectively is shown. While the GCN model achieves lower classification performance than either DLAB-VS or the SE(3)-Transformer model, the EGNN model performs better than DLAB-VS and the SE(3)-Transformer.

al. 2021) is a recently published equivariant graph neural network which achieves equivariance through a computationally efficient implementation described in 1.5.3. Using architectures previously evaluated on the QM9 data set (see section 5.3.5) I evaluated both of those models on the crystal structure data set (see Fig. 5.8). The GCN model performed worse than DLAB-VS or the SE(3)-Transformer model. The EGNN model outperformed all previously tested models, ranking the binding antibody in the top 2% and top 10% for 70.0% and 79.2% of antigen targets respectively. In the following experiments, I therefore used the EGNN model.

5.4.2 The EGNN model shows improvement over DLAB-VS on the model dataset

I trained a model architecture adapted with minimal changes from the EGNN paper (Satorras et al. 2021) on the model data set using the training set composition g-1 (see Table 5.1), using the same pose quality criteria as the best performing DLAB-VS model (see Table 5.2 and Fig. 5.9). This initial EGNN model, referred to in the following as EG-1, achieves better ranking performance than DLAB-VS on the model data set, ranking the binding antibody in the top 2% and top 10% for 5.2% and 17.1% of antigen targets, compared to DLAB-VS with 4.7% and 16.4%. Similar to DLAB-VS (see section 4.4.7), the performance of EG-1 can be improved by combining the EG-1 output scores with the ZDOCK output scores. This combined model (EG-1+ZDOCK) ranks the binding antibody in the top 2% and top 10% for 8.6% and 22.7% of antigen targets, compared to DLAB-VS+ZDOCK which achieves 6.4% and 19.7%.

5.4.3 Optimisation of the EGNN model increases classification performance

I next investigated whether optimising the EGNN architecture and training set composition could yield further improvements over the performance of EG-1. The results of this optimisation are listed in Table 5.2.

This process generated 5 main insights:

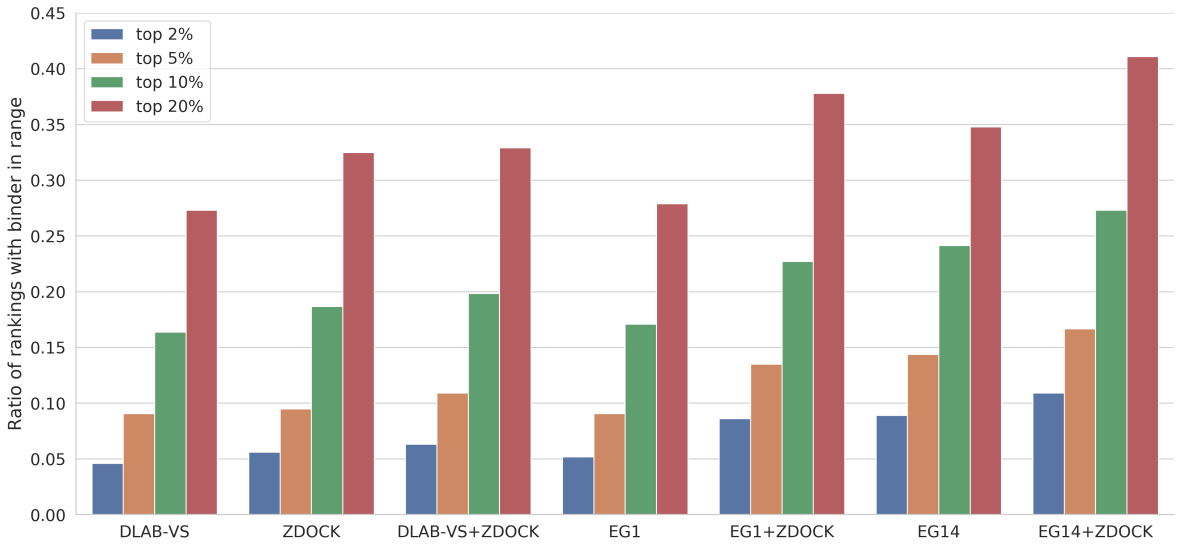

Figure 5.9: Comparison of DLAB-VS+ZDOCK and the EGNN models on the model data set. For comparison, I show the performance of the DLAB-VS ensemble, ZDOCK, and the DLAB-VS+ZDOCK ensemble described in the previous chapter. The non-optimised EGNN model (EG-1) slightly outperforms DLAB-VS and outperforms DLAB-VS+ZDOCK in combination with the ZDOCK scores. The optimised EGNN model described above (EG-14) outperforms EG1 both with and without the inclusion of ZDOCK scores.

ID	Graph mode	node type	layers	train set	other	% top ranked	+ZD
EG-1	int. nbhd	atoms	7x128	g-1		5.2%	8.6%
EG-2	int. nbhd	atoms	7x128	g-2		4.5%	6.5%
EG-3	int. nbhd	atoms	7x128	g-3		3.9%	5.7%
EG-4	int. nbhd	atoms	7x128	g-4		7.8%	8.9%
EG-5	int. nbhd	atoms	5x128	g-1		6.3%	7.0%
EG-6	int. nbhd	atoms	3x128	g-1		5.7%	8.3%
EG-7	int. edges	atoms	3x128	g-1		6.6%	9.5%
EG-8	int. nbhd	atoms	1x128	g-1		6.2%	8.0%
EG-9	int. nbhd	atoms	3x64	g-1		5.6%	7.8%
EG-10	int. nbhd	atoms	3x256	g-1		6.2%	8.0%
EG-11	int. nbhd	atoms	3x128	g-1	attn	5.6%	7.8%
EG-12	int. nbhd	atoms	3x128	g-4		8.3%	9.7%
EG-13	int. nbhd	atoms	3x128	g-4	no GN	8.8%	10.1%
EG-14	int. edges	atoms	3x128	g-4	no GN	**8.9%**	**10.9%**
EG-15	int. nbhd	residue	3x128	g-1		2.7%	3.7%
EG-16	int. edges	residue	3x128	g-1		6.2%	8.9%
EG-17	int. edges	residue	3x128	g-4		8.6%	10.2%

Table 5.2: Overview of the hyperparameter combinations used during optimisation of the EGNN model. For each set of parameters, the percentage of antigen targets for which the cognate antibody was top ranked by the EGNN model and by the ensemble of the EGNN model and ZDOCK (+ZD) is shown. The training set identifiers refer to the training set compositions listed in Table 5.1. No GN refers to model architectures not containing GraphNorm layers. EG-1 is the model used in the initial comparison to DLAB-VS, EG-14 is the optimised model used on the test sets.

Performance trends identified for DLAB-VS apply to the EGNN model. As previously observed during DLAB-VS training, the composition of the training set and particularly the cut-offs applied to poses labelled as binders are important determinants for model performance. During EGNN model optimisation, I again observed that lowering this cut-off from *fnat* 0.7 to *fnat* 0.5 (training set g-2, used to train model EG-2) lowers classification performance.

Including crystal structures in the training set increases classification performance. I investigated the impact of including crystal structures in the training set for a model trained to classify antibody-antigen binding using modelled antibody structures (training set g-4). Models trained using this training set (EG-4, EG-12, EG-14) performed better than the equivalent models trained on training set g-1. Since just training the EGNN model on the crystal structure data set and evaluating on the model data set (training set g-3) leads to inferior performance compared to both training sets g-1 and g-4, this increase in performance is due to the combination of modelled antibody structures and antibody crystal structures in the training set. The addition of docked crystal structures to the model data set during training improves performance by serving as a data augmentation strategy, providing additional high-quality data to the training set while also introducing realistic variance to the antibody confirmations seen by the model during training.

Simpler models are sufficient for good classification performance. In order to explore the viability of smaller and simpler models for the classification task (which would further increase the computational efficiency of the EGNN model), I investigated several model sizes (three to seven EGNN layers with 64, 128 and 256 channels each). Of the combinations tested in this

thesis, using three EGNN layers with 128 channels each resulted in similar performance to using seven EGNN channels and was used as basis for further parameter exploration (EG-6)[2]. This is a trend commonly observed in message passing GNNs, where deeper architectures can lead to over-smoothing (Cai and Wang 2020) and there is therefore a significant trade-off between gain of expressiveness from increased model depth and loss of expressiveness due to over-smoothing.

I further found that removing the recently published GraphNorm layer (Cai et al. 2020), which otherwise was included in all EGNN models except for EG-13 and EG-14, resulted in a further improvement in the model performance (from model EG-12 to EG-13). Though I did not further investigate this performance difference, the GraphNorm layer could have regularised the model too strongly, reducing overfitting at the cost of performance on the test set.

Attention confers less advantage than expected. Despite recent focus on attention-based architectures for equivariant graph neural networks, particularly in the context of protein structure (e.g. Jumper et al. (2021)), the addition of edge attention into the EGNN model (EG-11) did not result in improved performance over the attention-free model implementations. This was in line with the previous observation that the attention-free EGNN models outperformed the SE(3)-Transformer on the crystal data set (as well as on other benchmarks, see Satorras et al. (2021)).

[2]Later, I found that this is also the case for a model using just one EGNN layer (EG-8), though I did not explore this further in this thesis.

Graph generation can be simplified without performance loss. In order to test the graph generation schemes used to train the SE(3)-Transformer on the crystal structure data set for the EGNN models, I trained the EGNN models with both interaction neighbourhood and interaction edges graphs. Otherwise equivalent models performed better when trained using the interaction edges graphs (EG-6 *vs.* EG-7, EG-13 *vs.* EG-14, EG-15 *vs.* EG-16). Further, I implemented an additional graph generation scheme which reduced the number of graph nodes by only including Cα atoms and labelling these with a one-hot residue type vector instead of a one-hot atom type vector (see section 5.3.4.3). While this scheme failed to achieve classification performance using the interaction neighbourhood graphs (EG-15), when trained using interaction edges graphs (EG-16), this EGNN performed comparably to the equivalent model trained on atom type nodes (EG-7). Using only Cα atom nodes further reduced the number of pairwise distance calculations required during graph creation, reducing the impact of this computational bottleneck during training and testing.

This result differed from the results of similar experiments I conducted using the DLAB-VS architecture, where I found that reduced information content in the input data for the DLAB-VS model led to a deterioration of classification performance (see section 4.4.4). However, this result was in line with other studies using GNNs to predict protein structure or associated properties, in which residue-level graph representations were found to be sufficient to achieve good prediction performance (e.g. Eismann et al. (2020b), Abanades et al. (2021), Eismann et al. (2020a), and Jing et al. (2020)). Further, this result seemed to suggest that beyond the generation of an accurate docked pose, explicit representations of the amino acid side chains might not be necessary in order to perform accurate binding classification using GNN-based

approaches.

The optimised EGNN model (EG-14) represented a significant improvement over both EG-1 and DLAB-VS(+ZDOCK) (see Fig. 5.9): EG-14 ranked the binding antibody in the top 2% and top 10% for 8.9% and 24.1% of antigen targets, 10.9% and 27.3% when combined with the ZDOCK scores. In the following, I evaluated this best performing EGNN model (EG-14) on several held-out test sets and refer to it as DLAB-EG in line with the previous naming convention in this thesis.

5.4.4 The DLAB-EG model outperforms the previous DLAB pipeline on test sets

5.4.4.1 The DLAB-EG model outperforms DLAB-VS on the held-out post-snapshot test set

In order to accurately compare the performance of DLAB-EG with that of DLAB-VS and avoid multiple comparisons, I tested DLAB-EG after hyperparameter optimisation on the same held-out post-snapshot test set that DLAB-VS+ZDOCK was evaluated on in chapter 4 (see section 4.4.10). As in the previous chapter, I evaluated for this purpose a model ensemble comprised of all 40 models trained during the cross-validated training of DLAB-EG. The DLAB-EG+ZDOCK model showed improved classification performance over the DLAB-VS+ZDOCK model, both with and without the DLAB-Re thresholding approach described in section 4.3.3.4: on the post-snapshot data set without overlap to the model data set at 90% CDR sequence identity clustering, the binding antibody was ranked by DLAB-EG+ZDOCK in the top 2% and

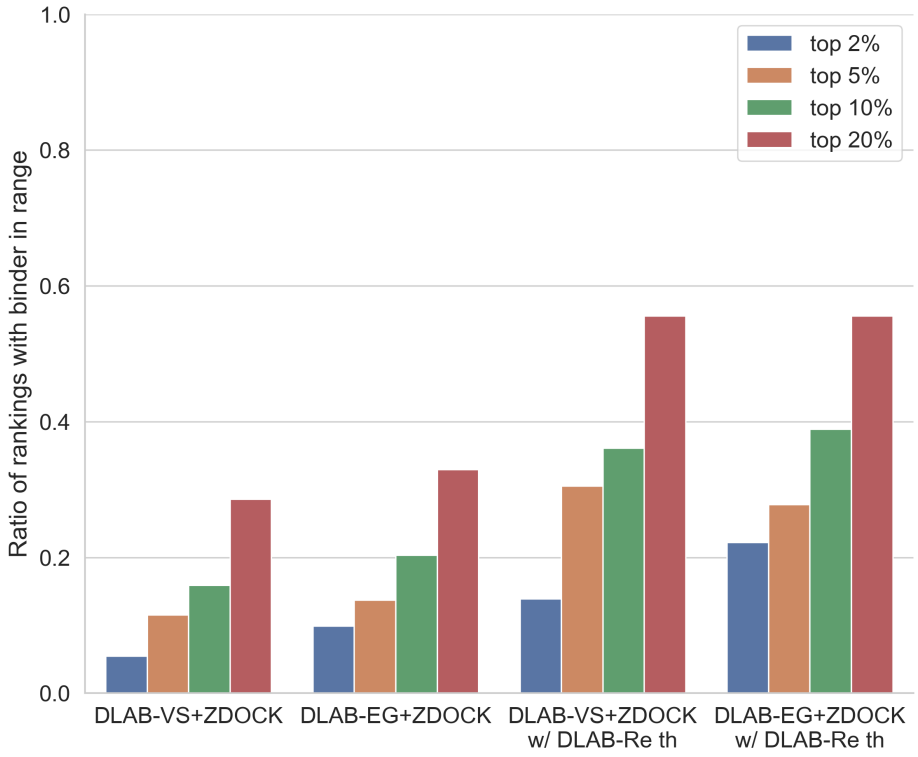

Figure 5.10: Performance of the DLAB-EG model on the held-out test set. For comparison, the performance of the DLAB-VS+ZDOCK model described in chapter 4 is displayed. The DLAB-EG model outperforms the DLAB-VS model, both before and after DLAB-Re score thresholding.

top 10% for 9.9% and 20.3% of antigen targets before DLAB-Re thresholding and 22.2% and 38.9% after DLAB-Re thresholding, as opposed to the DLAB-VS+ZDOCK model, which on the same data set achieved 5.5% and 15.9% before thresholding and 13.9% and 36.1% after thresholding (see Fig. 5.10). Other than the performance improvements observed during cross-validated model optimisation (see Fig. 5.9) the performance improvement over DLAB-VS+ZDOCK on the held-out test set was primarily seen at the top of the ranking, with the largest relative gain in binding antibodies ranked in the top 2%.

Like DLAB-VS+ZDOCK, DLAB-EG+ZDOCK further generalised well at stricter CDR sequence ID cut-offs for overlap between the held-out post-snapshot set and the training model data set (see Fig. 5.11).

5.4.4.2 DLAB-EG+ZDOCK performs similarly to DLAB-VS+ZDOCK on the epitope cluster test case

I further evaluated DLAB-EG+ZDOCK on the epitope cluster test case described in section 4.3.6.2. DLAB-VS+ZDOCK and DLAB-EG+ZDOCK performed similarly on this test case, with DLAB-EG+ZDOCK achieving marginally higher enrichment of binding antibodies in the top 10% and top 20% (see Fig. 5.12).

5.4.4.3 DLAB-VS+ZDOCK performs better than DLAB-EG+ZDOCK on the SARS-CoV-1/2 test case

On the SARS-CoV-1/2 test case, binders against the SARS-CoV-1 or SARS-CoV-2 have to be retrieved from a subset of an antibody model library (AML) whose members are either

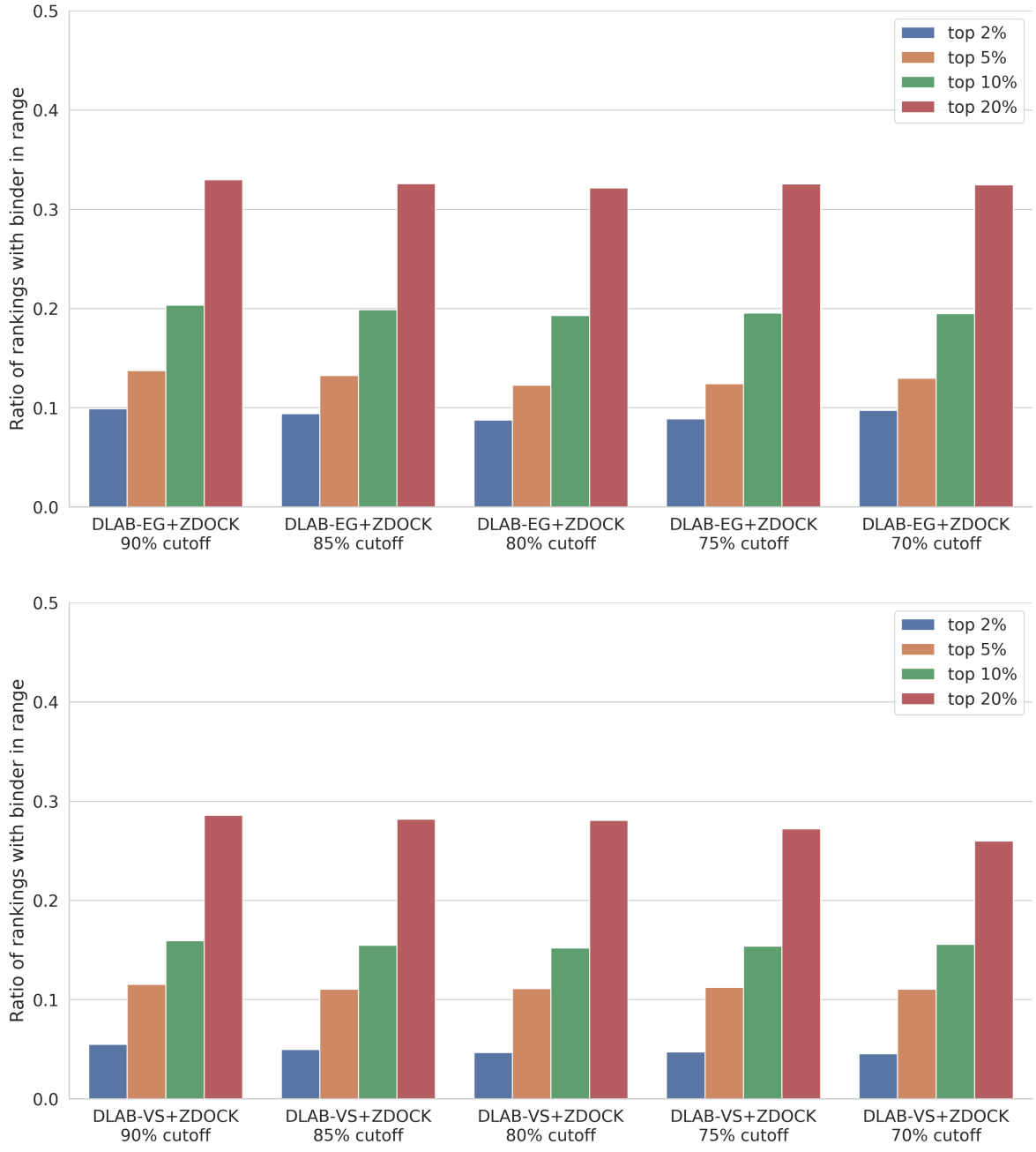

Figure 5.11: Influence of overlap with the training set on performance on the held-out test set for DLAB-EG (**Top**) and DLAB-VS (**Bottom**). Like DLAB-VS, DLAB-EG performance generalises to more stringent clustering cut-offs for allowed overlap to the training set.

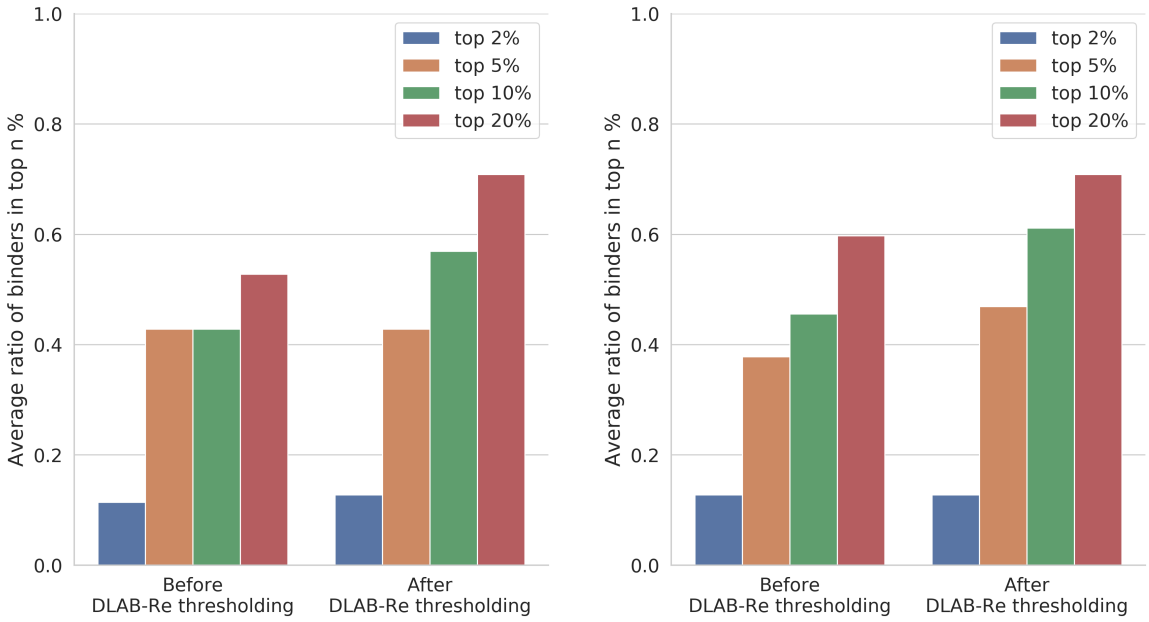

Figure 5.12: Performance of DLAB-VS (**Left**) and DLAB-EG (**Right**) on the epitope cluster test set. The two models perform similarly on the two clusters.

completely randomly chosen or based on length-matching the CDRs of the AML members to provide a more complex classification challenge (see section 4.3.6.3). Other than on the other test sets evaluated, the DLAB-VS+ZDOCK model performed considerably better than the DLAB-EG+ZDOCK model (see Fig. 5.13). This could potentially be explained by the fact that DLAB-VS and DLAB-EG have dissimilar model architectures and so have partially uncorrelated errors (see Fig. 5.14). While on most antigen targets, the DLAB-EG model outperforms the DLAB-VS model, this is not necessarily the case for all targets.

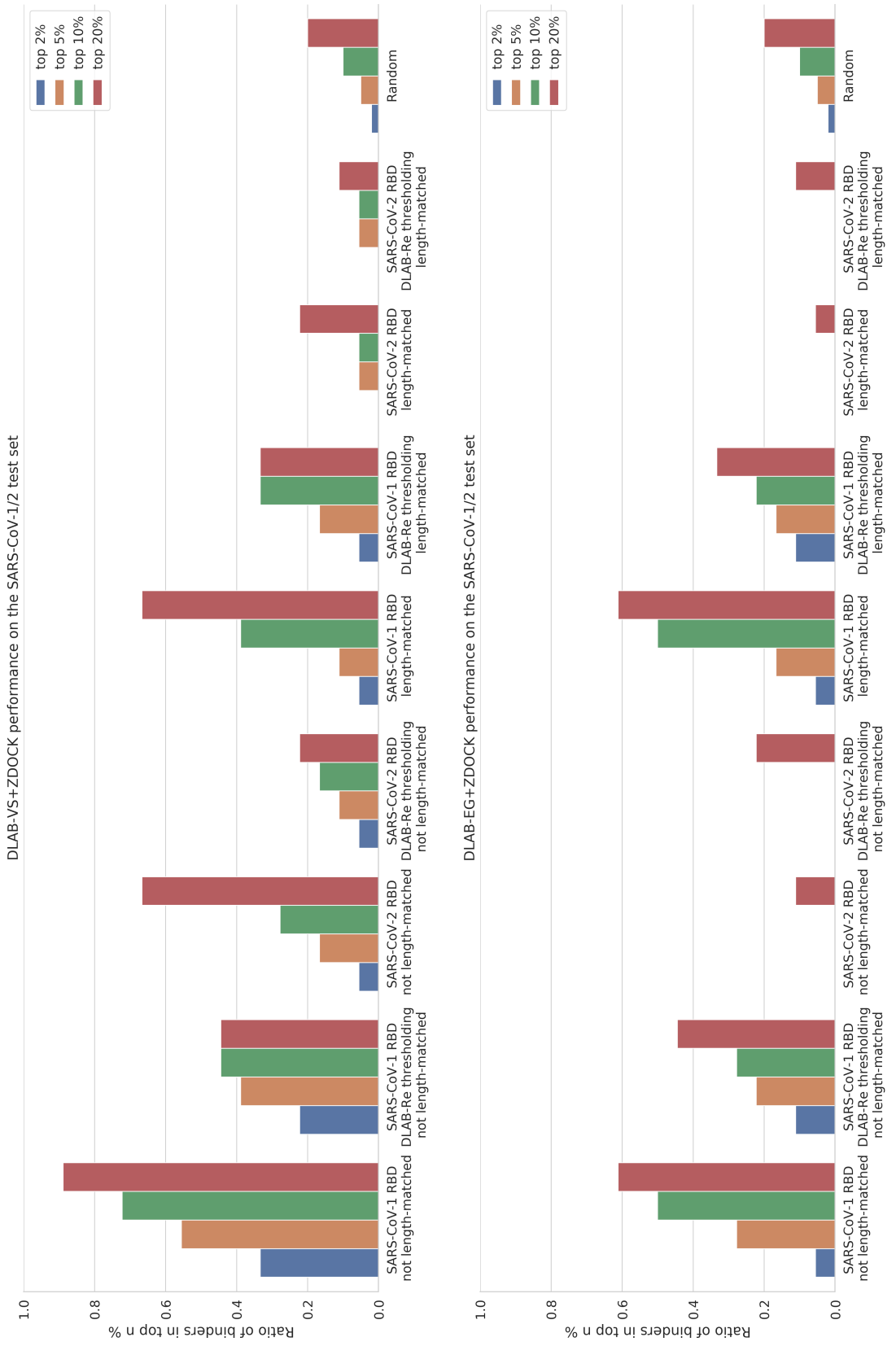

Figure 5.13: Performance of DLAB-VS+ZDOCK (**Top**) and DLAB-EG+ZDOCK (**Bottom**) on the SARS-CoV test set. Interestingly, the DLAB-EG model performed worse than the DLAB-VS model on this test case.

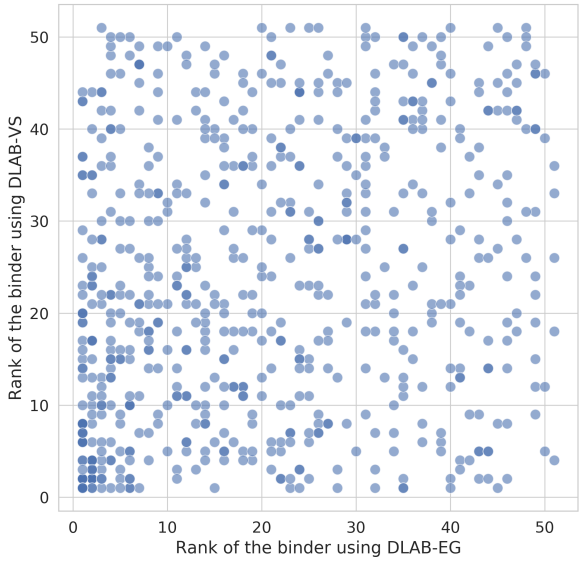

Figure 5.14: Scatter plot of ranking performance of DLAB-VS and DLAB-EG.

5.4.4.4 DLAB-EG+ZDOCK performs similarly to DLAB-VS+ZDOCK on the variant binding test set

I also tested DLAB-EG+ZDOCK on the same data set of SARS-CoV-2 binding antibodies with confirmed loss of binding against epitope variants (specified in Appendix table B.4) as DLAB-VS+ZDOCK, described in section 4.3.6.4. As for the DLAB-VS+ZDOCK model, for the 14 antibody-variant pairs in the data set, the DLAB-EG+ZDOCK score of the antibody-wild-type pair was higher than the score of the antibody-variant pair in 13/14 cases.

5.4.5 Attempts at docking-less classification were unsuccessful

Due to the noise introduced into the binding classification task by antibody-antigen docking (see chapter 3) and the comparatively slow speed of any docking algorithm (see section 1.3),

it would be beneficial to be able to perform binding classification in a docking-less scenario, where antibody and antigen structures are passed to the classifier separately and an output score is derived by combining independently generated feature vectors of antibody and antigen. I therefore implemented a model for this task, based on the SE(3)-Transformer and described in section 5.3.5.4 in order to investigate the potential of GNNs for this docking-less classification. In the initial experiments run for this thesis, using the model architecture described in 5.3.5.4 and Appendix table C.2 and training for up to ten epochs, the models used failed to converge during training and did not provide classification performance better than random on the crystal structure data set.

This failure was likely attributable to two factors. The model used was the worse performing SE(3)-Transformer which due to its high computational cost during training and inference could not be trained using an exhaustive hyperparameter search. Secondly, the model architecture was comparatively simple and did not use any information about the correct binding interface during training, nor did it incorporate useful biases inherent to protein-protein interaction, which have recently been used to generate a GNN-based rigid-body docking model (Ganea et al. 2021). I discuss potential future research towards this goal in chapter 6.

5.4.6 Exposing several docked poses to the network at once enables training on the raw model dataset

In the previous chapter, I demonstrated that combining the classification scores for several input interaction poses can improve model performance. In order to directly harness this insight in

an end-to-end learnable manner and create a classification model which can be trained on data of the same quality on which the model makes predictions at test time, I implemented a multi-pose EGNN architecture, which in each forward pass through the model extracts graph feature vectors from the top ten ranked poses for an antibody-antigen pairing, pools them and uses that joint representation to derive a classification score (see section 5.3.5.5 and Fig. 5.6).

I trained this architecture on the same cross-validation splits as the DLAB-EG architecture, testing the influence of different training set compositions as well as different graph generation setups (see Tables 5.1 and 5.2). While none of the hyperparameter combinations tested in this initial analysis achieved performance comparable to DLAB-EG, the trends observed during development of DLAB-EG held for this model setup: models trained on dataset tp-b, which combined antibody crystal structures and modelled antibody structures as well as models using interaction edge graphs performed better during cross-validation. The best performing ten-pose model, TP-2, achieved higher performance on the model data set than DLAB-VS (see Fig. 5.15) and, in combination with ZDOCK, performed similarly to DLAB-VS+ZDOCK on the held-out post-snapshot test set (see Fig. 5.16), indicating that there is further scope for optimisation for this model type.

A drawback of the ten-pose model architecture is the comparatively slow training time. Each example, including non-binding antibody-antigen pairings, required ten forward passes through the GNN layers of the model, leading to a ten-fold increase in training for the same number of epochs while the model at the same time also required lower learning rates to converge and thus needed to train for 40 epochs (as opposed to 20 epochs for the EGNN models trained above).

ID	Graph mode	node type	layers	training set	other	% top ranked
TP-1	int. edges	residues	3x128	tp-a	shuffled	4.3%
TP-2	int. edges	residues	3x128	tp-b	shuffled	5.9%
TP-3	int. nbhd	atoms	1x128	tp-b	shuffled	4.0%
TP-4	int. nbhd	atoms	3x128	tp-a		3.5%

Table 5.3: Overview of the cross-validation results on the multi-pose models. All models share the general architecture shown in Fig. 5.6. Shuffled refers to randomising the order of the ten input poses at train time. Training set compositions a and b refer to the training set setups described in Table 5.1.

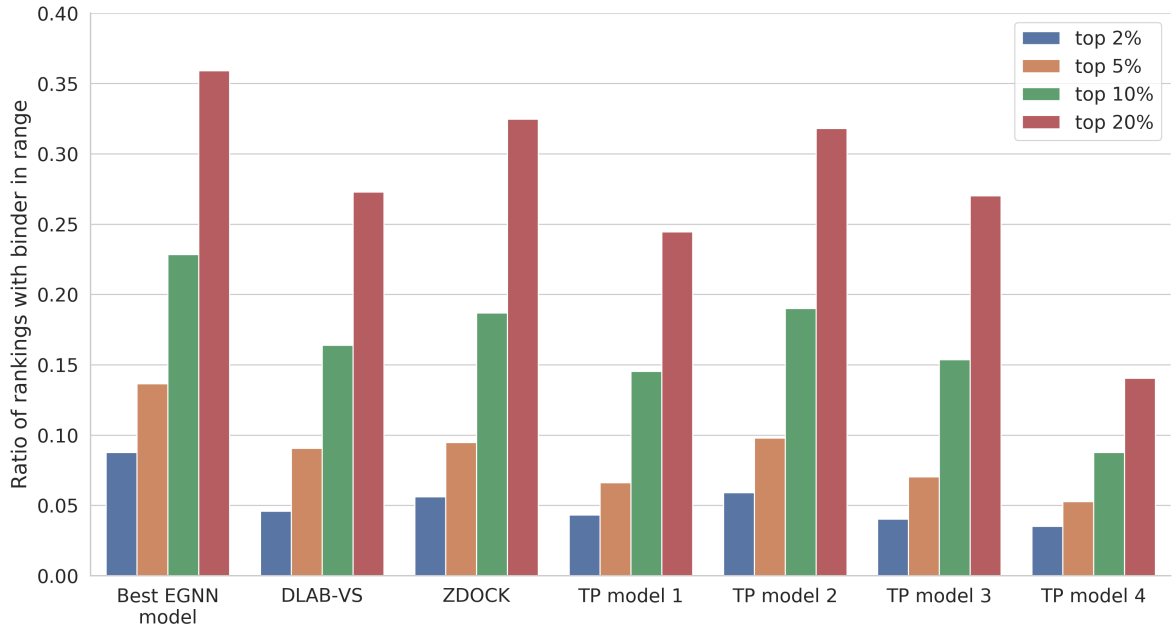

Figure 5.15: Comparison of the performance of different multi-pose EGNN architectures. While the best ten-pose model (TP model 2) tested in this thesis outperforms DLAB-VS, it performs worse than the optimised single pose EGNN model. In order to simplify model comparisons, the performance of the combined models (like DLAB-VS+ZDOCK) is not shown.

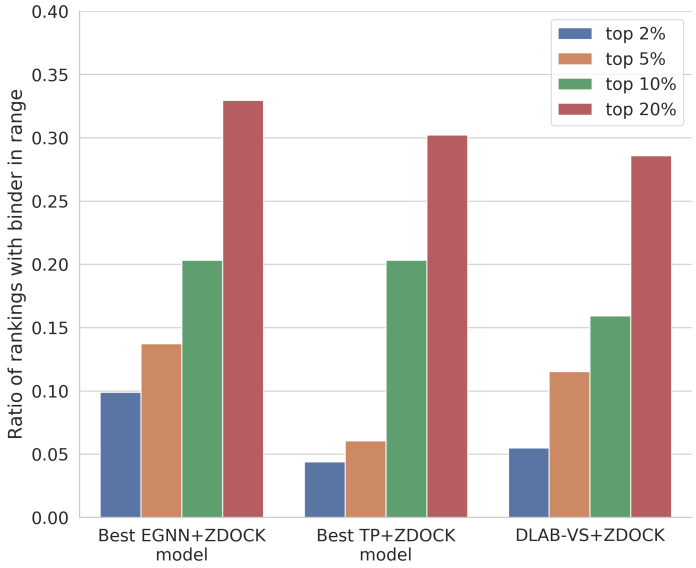

Figure 5.16: Performance of the best ten-pose EGNN model (TP-2) combined with the ZDOCK output on the held-out test set. The best ten-pose model performs worse than the optimised EGNN+ZDOCK model (DLAB-EG).

5.5 Discussion

In this chapter, I have presented significant improvements over the DLAB-VS classification model by employing state-of-the-art equivariant graph neural networks.

I have shown that the E(n)-equivariant graph convolutional neural network architecture of Satorras et al. (2021) can be trained on antibody-antigen structural data to achieve performance improvements over DLAB-VS, which can be further increased through optimisation of the architecture itself as well as the input graph generation algorithm. DLAB-EG, the optimised EGNN model, almost doubled the ratio of antigens for which the binding antibody was ranked in the top 2% compared to DLAB-VS and outperformed the DLAB-VS model on relevant test

sets. I further showed that the improvements made to the DLAB-VS model (the combination with ZDOCK output scores and DLAB-Re thresholding) similarly improve prediction for the DLAB-EG model.

Beyond the increased performance, using graph neural networks also allowed me to significantly simplify the data input pipeline for the classification models. DLAB-EG, unlike DLAB-VS, does not rely on pre-generation of input files for each docking pose (reducing storage requirements) while also making use of efficient distance calculation algorithms and a fast model architecture to keep computational cost low at run time.

There is further scope for optimisation of the DLAB-EG model as the hyperparameter optimisation I conducted in this chapter was not exhaustive. The analysis detailed in Table 5.2 shows that further performance improvement could be achieved using several promising avenues. Smaller models with depth under three layers should be explored in more detail. Additional graph creation schemes beyond the three schemes should also be evaluated, including the use of self-loops connecting nodes to themselves, central nodes allowing efficient information flow through the graph and graph schemes set between the interaction neighbourhood and interaction edge graphs in terms of complexity, incorporating ”intra" edges in the interaction edge graph. Further, the potential of the simplified residue node graphs, which, using similar implementations, have been used successfully in previous publications for the prediction of protein complex structures (Eismann et al. 2020a), should be evaluated further.

There is also scope to extend the training set used for DLAB-EG. In this chapter, I used the same data sets I originally created to train DLAB-VS (in 2018), in order to conduct a fair

comparison between the two model types. However, as has been shown in many studies (for example by Imrie et al. (2018)), providing more diverse data during training is likely to improve the performance of the DLAB-EG model. Thus, further exploration of this model type should be conducted on updated data sets. Since the creation of the post-snapshot test set, 313 non-redundant antibody structures in complex with antigens have been added to SAbDab. This means that the available data for training of structure-based antibody-antigen binding classification models has increased by more than 50%.

Additionally, I demonstrated in this chapter that multi-pose models enable training on the full set of modelled antibody-antigen docked poses without applying pose quality cut-offs. As well as having ample scope for optimisation of the architecture itself, the multi-pose architecture introduced in this chapter could therefore be particularly well suited to benefit from the increased amount of data which could be included in the training set.

DLAB-EG presents a significant improvement over DLAB-VS for antibody-antigen binding classification. The superior performance achieved by DLAB-EG demonstrates that, in line with several recent publications in this field (e.g. Abanades et al. (2021), Jing et al. (2020), Jumper et al. (2021), and Eismann et al. (2020a)), graph neural networks are well suited for antibody virtual screening. In the near future, DLAB-EG could be further improved using more comprehensive structural data sets and additional exploration of model architecture choices.

6 | Discussion and Future Work

Contents

6.1 Discussion

In this thesis, I have described significant methodological advances for the structure-based analysis of antibody structures, employing deep learning methods across the binding prediction pipeline for both docking pose assessment (chapter 3) and binding prediction (chapters 4 and 5). I have demonstrated the applicability of modern deep learning methods for the prediction of antibody-antigen binding and have shown that methodological insights from both machine learning based small molecule virtual screening and protein structure prediction can be translated into improvements for the antibody domain. The research detailed in this thesis charts a path for further improvement of structure-based antibody property prediction and the

integration of deep learning approaches into a fully *in-silico* pipeline for the identification of binding antibodies against a specific epitope from antibody model libraries.

6.1.1 Bespoke data sets can be created using SAbDab

In chapter 2, I detailed several improvements to the Structural Antibody Database (SAbDab), which will enable the creation of structure data sets specific to the analysis of defined target antigens (through enhanced search features and links to ancillary databases) or antibody formats (through the curation of SAbDab-nano). These features will in the future enable fine-tuning virtual screening pipelines to narrow applications, such as the prediction of disease specific antibody binders.

6.1.2 CNN models can improve antibody-antigen docking

In chapter 3, I have demonstrated that antibody-antigen docked pose ranking and docking quality assessment can be significantly improved through CNN-based deep learning approaches. I showed that reranking and assessment of docked poses of computationally generated antibody structures requires models trained specifically on the antibody domain and on computationally generated structures. The model developed for this purpose, DLAB-Re, improves ranking of antibody-antigen docked poses and enables the identification of well docked antibody-antigen pairings. Future work in this area should revisit the direct regression of pose quality metrics, which was unsuccessful in chapter 3, and evaluate the performance of GNN approaches on these tasks. These models have already been demonstrated to improve performance for the assessment of general protein-protein complex poses (Wang et al. 2021a; Réau et al. 2021) as

well as for antibody-antigen binding classification (see below).

6.1.3 Antibody-antigen binding can be predicted using CNN models

Building upon the work in chapter 3, in chapter 4 I extended the structure-based CNN approach to the prediction of antibody-antigen binding, creating DLAB-VS, the first structure-based deep learning tool for antibody virtual screening. I demonstrated that binding prediction from computationally generated antibody structures is challenging but can be achieved using methodological insights from small molecule virtual screening and model ensembles. I demonstrated that binding prediction is dependent on both antibody model quality and docking performance and validated the tool on several use cases, providing a proof-of-principle for an antibody model library-based virtual screening pipeline.

6.1.4 GNN models enable improvements to antibody virtual screening

Motivated by recent developments in machine learning and protein structure prediction research, in chapter 5 I improved the initial DLAB-VS model by switching from a CNN-based approach to a GNN-based approach, demonstrating the superiority of GNN approaches for structural analysis on the antibody domain and generating insights into performance determining factors. Recently published equivariant message passing networks outperformed simpler approaches while also providing fast inference and requiring less hard-drive storage than the DLAB-VS CNN model. I have so far demonstrated the performance improvement of switching from CNN-based models to GNN-based model only on the binding-prediction step, but not on the docking pose assessment task. However, as mentioned in section 6.1.2, it is reasonable

to suggest that improvements observed in other studies for docking pose ranking using GNN-based approaches (Wang et al. 2021a; Eismann et al. 2020a; Réau et al. 2021) will similarly apply here. In both chapter 4 and chapter 5, I have demonstrated that accurate classification of antibody-antigen binding using structural machine learning approaches is strongly correlated with the quality of computationally generated antibody structures. Further research on these models should therefore incorporate recent improvements in antibody modelling using end-to-end machine learning (Abanades et al. 2021).

6.2 Future work

There are several promising avenues for the continuation of the work laid out in this thesis, which I detail below in order of increasing complexity, based on the amount of novel model implementation that would be necessary. Obviously, neither of the suggested expansions of the DLAB framework are mutually exclusive and future work could be a combination of the suggested approaches.

6.2.1 Transfer learning

An obvious immediate research direction to improve the DLAB pipeline is the use of transfer learning. Using general protein-protein interaction data for initial model training and fine-tuning on the antibody domain or jointly training models on both domains has been shown to improve prediction performance on epitope-prediction models (Hou et al. 2021; Pittala and Bailey-Kellogg 2020; Dai and Bailey-Kellogg 2021). The recently compiled databases of gen-

eral protein-protein interaction structures (Townshend et al. 2019; Morehead et al. 2021) would provide a more than 40-fold increase in the size of the available training set.

6.2.2 Evaluation on synthetic data sets

Secondly, while experimental validation of virtual screening pipelines would be time-intensive, additional evaluation of the models developed in this thesis could be undertaken using the synthetic data generation approach laid out by Robert et al. (2021). While adapting the CNN-based DLAB models could prove challenging in this context (due to the difference between atom-level and residue-level grid-based models), graph-based models are easily translatable into a synthetic framework (as they do not require fixed-size input and only implicitly use Euclidean geometry), which would enable additional optimisation of DLAB-EG in a high-data scenario. Here, two main avenues of research could generate useful insights: Training and evaluation in this simulated high-data scenario would enable the identification of model types and architectures likely to be well-performing once larger amounts of experimentally determined structure are available for the creation of training data sets. Another interesting approach would be to investigate if model performance could be improved by first pre-training models on large volumes of simulated structural data before fine-tuning on the data sets based on experimentally determined structures used in this thesis.

6.2.3 Joint learning on sequence and structure

Recently, significant progress has been made on applying language embedding models to protein sequences (e.g. Brandes et al. 2021), as well as joint embeddings of protein sequence and

structure (e.g. Mansoor et al. (2021)). Using combined embeddings of sequence and structure has already been demonstrated to improve paratope prediction (Del Vecchio et al. 2021) and could be integrated into virtual screening models. These joint embedding approaches are yet to be applied to the analysis of complex structures of interacting proteins and thus would require adaption to this domain, for example when considering the representation of gaps in discontinuous paratope/epitope segments (see also Akbar et al. (2021a)).

6.2.4 End-to-end learning

Lastly, the emergence of AlphaFold2 and similar methods for the prediction and analysis of protein structures has demonstrated that machine learning methods which can be trained end-to-end, learning and optimising across the entire structure generation and analysis pipeline, can outperform approaches in which the deep learning frameworks only contribute parts of the pipeline. While an initial attempt towards this goal in chapter 5 did not yield an improvement over the existing pipeline, recent publications have shown that both antibody modelling (Abanades et al. 2021) and rigid-body-docking (Ganea et al. 2021) are accessible to graph-based deep learning models. A successful implementation of an end-to-end antibody virtual screening model could therefore not only improve the prediction performance of structure-based antibody virtual screening, but also eliminate the biggest computational bottleneck in the pipeline in its current form: ZDOCK based rigid-body docking of one antibody-antigen pair requires approximately 300s of CPU compute time, compared to approximately 0.03s for the generation of a graph from a docked pose and negligible time to run DLAB-EG on the input graph. Initially, this could take the form of a two-step model, which would first predict

the transformation matrix for the structures of the interaction partners via rigid-body docking, which would then be used to generate the complex pose and pass it along to a binding prediction module like DLAB-EG. If this setup proves successful, it could be expanded to compute solely on latent space representations of antibody and antigen to avoid re-computation of interaction graphs.

6.2.5 Community evaluation of predictive models

It is evident that the immediate future of the antibody property prediction field, both for structure- and sequence-based analysis, lies in deep learning methods, with some form of geometrical deep learning likely to be the method of choice for structure-based antibody analysis for some time. This has already led to a flurry of novel methods published over the last two years. Like the protein structure prediction field, the field of machine learning models for the prediction of antibody properties would now benefit immensely from a community-wide independent assessment and benchmarking effort, requiring the precise definition of unsolved tasks and the compilation of unbiased benchmarking data sets. Such a CASP-like community effort would enable the fast emergence of the best ideas from the diverse approaches currently suggested and lead to the generation of standardised, high quality data sets for model training and evaluation.

6.3 Final remarks

This is an extremely exciting time for structure-based antibody research. Recent successes in antibody structure prediction suggest that the full space of antibody structures will soon be

quickly predictable with high accuracy, while the work detailed in this thesis lays out a roadmap for the utilisation of these models for antibody therapeutics discovery.

The DLAB pipeline described in this thesis provides a significant contribution to the emerging field of structure-based machine learning models for antibody virtual screening. Recently published improvements in antibody and antigen structural modelling and docking algorithms will expand the capabilities of DLAB and pipelines like it in the future. While additional optimisation and experimental validation will be necessary before the adoption of these virtual screening approaches into discovery campaigns, the research I conducted over the course of this thesis serves as a proof-of-concept for the application of modern deep learning approaches to antibody-antigen binding prediction. The fully *in-silico* identification of therapeutic antibody candidates from antibody model libraries is becoming increasingly possible and the contributions made to data set creation as well as deep learning-based docking evaluation and virtual screening of computationally generated antibody structures present an important step towards the realisation of the potential of this screening pipeline.

Bibliography

Abanades, Brennan et al. (2021). "ABlooper: Fast accurate antibody CDR loop structure prediction with accuracy estimation". In: *bioRxiv*. DOI: 10.1101/2021.07.26.453747 (cit. on pp. 25, 26, 30, 55, 104, 115, 162, 174, 189, 202, 206, 208).

Abhinandan, K.R. and Andrew C.R. Martin (2008). "Analysis and improvements to Kabat and structurally correct numbering of antibody variable domains". In: *Molecular Immunology* 45.14, pp. 3832–3839. DOI: 10.1016/j.molimm.2008.05.022 (cit. on pp. 13, 14).

Adolf-Bryfogle, Jared et al. (2018). "RosettaAntibodyDesign (RAbD): A general framework for computational antibody design". In: *PLoS Computational Biology* 14.4, e1006112. DOI: 10.1371/journal.pcbi.1006112 (cit. on pp. 44, 45, 119, 122).

Akbar, Rahmad et al. (2021a). "A compact vocabulary of paratope-epitope interactions enables predictability of antibody-antigen binding". In: *Cell Reports* 34.11, p. 108856. DOI: 10.1016/j.celrep.2021.108856 (cit. on pp. 62, 120, 208).

Akbar, Rahmad et al. (2021b). "In silico proof of principle of machine learning-based antibody design at unconstrained scale". In: *bioRxiv*. DOI: 10.1101/2021.07.08.451480 (cit. on p. 62).

Alkan, Sefik S. (2004). "Monoclonal antibodies: the story of a discovery that revolutionized science and medicine". In: *Nature Reviews Immunology* 4.2, pp. 153–156. DOI: 10.1038/nri1265 (cit. on p. 40).

Allcorn, L. C. and A. C. R. Martin (2002). "SACS–Self-maintaining database of antibody crystal structure information". In: *Bioinformatics* 18.1, pp. 175–181. DOI: 10.1093/bioinformatics/18.1.175 (cit. on p. 21).

Almagro, Juan C. et al. (2014). "Second Antibody Modeling Assessment (AMA-II)". In: *Proteins: Structure, Function and Bioinformatics* 82.8, pp. 1553–1562. DOI: 10.1002/prot.24567 (cit. on p. 30).

Almagro, Juan C. et al. (2018). "Progress and challenges in the design and clinical development of antibodies for cancer therapy". In: *Frontiers in Immunology* 8, p. 1751. DOI: 10.3389/fimmu.2017.01751 (cit. on p. 118).

AlQuraishi, Mohammed (2019). "End-to-End Differentiable Learning of Protein Structure". In: *Cell Systems* 8.4, 292–301.e3. DOI: 10.1016/j.cels.2019.03.006 (cit. on p. 25).

Ambrosetti, F. et al. (2019). "Information-Driven Modelling of Antibody-Antigen Complexes". In: *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3362436 (cit. on pp. 35, 88, 115).

Ambrosetti, Francesco et al. (2020). "proABC-2: PRediction of AntiBody contacts v2 and its application to information-driven docking". In: *Bioinformatics* 36.20, pp. 5107–5108. DOI: 10.1093/bioinformatics/btaa644 (cit. on p. 35).

Baek, Minkyung et al. (2021). “Accurate prediction of protein structures and interactions using a three-track neural network”. In: *Science* 373.6557, pp. 871–876. DOI: 10.1126/science.abj8754 (cit. on pp. 25, 171).

Bahai, Akash et al. (2021). “EpitopeVec: linear epitope prediction using deep protein sequence embeddings”. In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btab467 (cit. on p. 43).

Barré, Samantha et al. (1994). “Structural conservation of hypervariable regions in immunoglobulins evolution”. In: *Nature Structural and Molecular Biology* 1 (12), pp. 915–920. DOI: 10.1038/nsb1294-915 (cit. on p. 85).

Bates, Adam and Christine A. Power (2019). “David vs. Goliath: The Structure, Function, and Clinical Prospects of Antibody Fragments”. In: *Antibodies* 8.2, p. 28. DOI: 10.3390/antib8020028 (cit. on pp. 16, 18).

Bauer, Matthias R. et al. (2013). “Evaluation and Optimization of Virtual Screening Workflows with DEKOIS 2.0 - A Public Library of Challenging Docking Benchmark Sets”. In: *Journal of Chemical Information and Modeling* 53.6, pp. 1447–1462. DOI: 10.1021/ci400115b (cit. on p. 57).

Beck, Alain et al. (2017). “Strategies and challenges for the next generation of antibody-drug conjugates”. In: *Nature Reviews Drug Discovery* 16.5, pp. 315–337. DOI: 10.1038/nrd.2016.268 (cit. on p. 18).

Benjin, Xu and Liu Ling (2020). “Developments, applications, and prospects of cryo-electron microscopy”. In: *Protein Science* 29.4, pp. 872–882. DOI: 10.1002/pro.3805 (cit. on p. 19).

Berman, Helen M. et al. (2000). “The Protein Data Bank”. In: *Nucleic Acids Research* 28.1, pp. 235–242. DOI: 10.1093/nar/28.1.235 (cit. on pp. 21, 67, 85).

Binshtein, Elad and Melanie D. Ohi (2015). “Cryo-Electron Microscopy and the Amazing Race to Atomic Resolution”. In: *Biochemistry* 54.20, pp. 3133–3141. DOI: 10.1021/acs.biochem.5b00114 (cit. on p. 20).

Boyles, Fergus, Charlotte M. Deane, and Garrett M Morris (2019). “Learning from the ligand: using ligand-based features to improve binding affinity prediction”. In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btz665 (cit. on p. 43).

Bradbury, Andrew R. M. and Andreas Plückthun (2015). “Reproducibility: Standardize antibodies used in research”. In: *Nature* 518.7537, pp. 27–29. DOI: 10.1038/518027a (cit. on p. 40).

Bradbury, Andrew R. M. et al. (2018). “When monoclonal antibodies are not monospecific: Hybridomas frequently express additional functional variable regions”. In: *mAbs* 10.4, pp. 539–546. DOI: 10.1080/19420862.2018.1445456 (cit. on p. 40).

Brandes, Nadav et al. (2021). "ProteinBERT: A universal deep-learning model of protein sequence and function". In: *bioRxiv*. DOI: 10.1101/2021.05.24.445464 (cit. on pp. 58, 207).

Briney, Bryan et al. (2019). "Commonality despite exceptional diversity in the baseline human antibody repertoire". In: *Nature* 566.7744, pp. 393–397. DOI: 10.1038/s41586-019-0879-y (cit. on p. 4).

Brown, Tom et al. (2020). "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., pp. 1877–1901 (cit. on p. 48).

Bryant, P et al. (2021). "Improved prediction of protein-protein interactions using AlphaFold2 and extended multiple-sequence alignments". In: *bioRxiv*. DOI: 10.1101/2021.09.15.460468 (cit. on pp. 24, 61).

Bujotzek, Alexander et al. (2015). "Prediction of VH-VL domain orientation for antibody variable domain modeling". In: *Proteins: Structure, Function and Bioinformatics* 83.4, pp. 681–695. DOI: 10.1002/prot.24756 (cit. on p. 119).

Cai, Chen and Yusu Wang (2020). "A Note on Over-Smoothing for Graph Neural Networks". In: *arXiv*, arXiv:2006.13318 (cit. on p. 188).

Cai, Tianle et al. (2020). "GraphNorm: A Principled Approach to Accelerating Graph Neural Network Training". In: *arXiv*, arXiv:2009.03294 (cit. on p. 188).

Callaway, Ewen (2015). "The revolution will not be crystallized: a new method sweeps through structural biology". In: *Nature* 525.7568, pp. 172–174. DOI: 10.1038/525172a (cit. on p. 19).

Cavalli, A. et al. (2007). "Protein structure determination from NMR chemical shifts". In: *Proceedings of the National Academy of Sciences* 104.23, pp. 9615–9620. DOI: 10.1073/pnas.0610313104 (cit. on p. 20).

Chen, Lieyang et al. (2019). "Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening". In: *PLOS ONE* 14.8, e0220113. DOI: 10.1371/journal.pone.0220113 (cit. on p. 57).

Chen, Xingyao et al. (2020). "Predicting Antibody Developability from Sequence using Machine Learning". In: *bioRxiv*. DOI: 10.1101/2020.06.18.159798 (cit. on p. 45).

Cheng, Yifan et al. (2015). "A Primer to Single-Particle Cryo-Electron Microscopy". In: *Cell* 161.3, pp. 438–449. DOI: 10.1016/j.cell.2015.03.050 (cit. on p. 20).

Chiu, Mark L and Gary L Gilliland (2016). "Engineering antibody therapeutics". In: *Current Opinion in Structural Biology* 38, pp. 163–173. DOI: 10.1016/j.sbi.2016.07.012 (cit. on p. 119).

Choi, Yoonjoo and Charlotte M. Deane (2010). "FREAD revisited: Accurate loop structure prediction using a database search algorithm". In: *Proteins: Structure, Function, and Bioinformatics* 78.6, pp. 1431–1440. DOI: 10.1002/prot.22658 (cit. on p. 29).

Chothia, Cyrus and Arthur M. Lesk (1987). "Canonical structures for the hypervariable regions of immunoglobulins". In: *Journal of Molecular Biology* 196.4, pp. 901–917. DOI: 10.1016/0022-2836(87)90412-8 (cit. on p. 13).

Chowdhury, Ratul, Matthew F. Allan, and Costas D. Maranas (2018). "OptMAVEn-2.0: De novo Design of Variable Antibody Regions Against Targeted Antigen Epitopes". In: *Antibodies* 7.3, p. 23. DOI: 10.3390/antib7030023 (cit. on pp. 44, 45, 119).

Cock, Peter J. A. et al. (2009). "Biopython: freely available Python tools for computational molecular biology and bioinformatics". In: *Bioinformatics* 25.11, pp. 1422–1423. DOI: 10.1093/bioinformatics/btp163 (cit. on pp. 87, 122, 130).

Cook, Charles E. et al. (2016). "The European Bioinformatics Institute in 2016: Data growth and integration". In: *Nucleic Acids Research* 44.D1, pp. D20–D26. DOI: 10.1093/nar/gkv1352 (cit. on p. 19).

Dai, Bowen and Chris Bailey-Kellogg (2021). "Protein interaction interface region prediction by geometric deep learning". In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btab154 (cit. on pp. 44, 58, 59, 162, 206).

Del Alamo, Diego et al. (2021). "Sampling the conformational landscapes of transporters and receptors with AlphaFold2". In: *bioRxiv*. DOI: 10.1101/2021.11.22.469536 (cit. on p. 24).

Del Vecchio, Alice et al. (2021). "Neural message passing for joint paratope-epitope prediction". In: *arXiv*, arXiv:2106.00757 (cit. on pp. 59, 208).

Desmyter, Aline et al. (2001). "Antigen Specificity and High Affinity Binding Provided by One Single Loop of a Camel Single-domain Antibody". In: *Journal of Biological Chemistry* 276.28, pp. 26285–26290. DOI: 10.1074/jbc.M102107200 (cit. on p. 17).

Desta, Israel T. et al. (2020). "Performance and Its Limits in Rigid Body Protein-Protein Docking". In: *Structure* 28.9, 1071–1081.e3. DOI: 10.1016/j.str.2020.06.006 (cit. on pp. 33, 36, 84, 116).

Deszyński, Piotr et al. (2021). "INDI - Integrated Nanobody Database for Immunoinformatics". In: *medRxiv*. DOI: 10.1101/2021.08.04.21261581 (cit. on p. 78).

Dominguez, Cyril, Rolf Boelens, and Alexandre M.J.J. Bonvin (2003). "HADDOCK: A protein-protein docking approach based on biochemical or biophysical information". In: *Journal of the American Chemical Society* 125.7, pp. 1731–1737. DOI: 10.1021/ja026939x (cit. on p. 35).

Dondelinger, Mathieu et al. (2018). "Understanding the Significance and Implications of Antibody Numbering and Antigen-Binding Surface/Residue Definition." In: *Frontiers in immunology* 9, p. 2278. DOI: 10.3389/fimmu.2018.02278 (cit. on pp. 12, 13).

Drago, Joshua Z., Shanu Modi, and Sarat Chandarlapaty (2021). "Unlocking the potential of antibody-drug conjugates for cancer therapy". In: *Nature Reviews Clinical Oncology* 18.6, pp. 327–344. DOI: 10.1038/s41571-021-00470-8 (cit. on p. 18).

Dunbar, James and Charlotte M. Deane (2015). "ANARCI: Antigen receptor numbering and receptor classification". In: *Bioinformatics* 32.2, pp. 298–300. DOI: 10.1093/bioinformatics/btv552 (cit. on pp. 14, 85, 130).

Dunbar, James et al. (2013). "ABangle: Characterising the VH-VL orientation in antibodies". In: *Protein Engineering, Design and Selection* 26.10, pp. 611–620. DOI: 10.1093/protein/gzt020 (cit. on pp. 6, 29).

Dunbar, James et al. (2014). "SAbDab: The structural antibody database". In: *Nucleic Acids Research* 42.D1, pp. D1140–6. DOI: 10.1093/nar/gkt1043 (cit. on pp. 19–22, 44, 62, 66, 67, 69, 82, 85).

Ehrenmann, François, Quentin Kaas, and Marie-Paule Lefranc (2010). "IMGT/3Dstructure-DB and IMGT/DomainGapAlign: a database and a tool for immunoglobulins or antibodies, T cell receptors, MHC, IgSF and MhcSF". In: *Nucleic Acids Research* 38.suppl_1, pp. D301–D307. DOI: 10.1093/nar/gkp946 (cit. on pp. 21, 78).

Eismann, Stephan et al. (2020a). "Hierarchical, rotation-equivariant neural networks to select structural models of protein complexes". In: *Proteins: Structure, Function, and Bioinformatics*, prot.26033. DOI: 10.1002/prot.26033 (cit. on pp. 55, 60, 84, 162, 163, 189, 201, 202, 206).

Eismann, Stephan et al. (2020b). "Protein model quality assessment using rotation-equivariant, hierarchical neural networks". In: *arXiv*, arXiv:2011.13557 (cit. on pp. 55, 189).

Emmerich, Christoph H. et al. (2021). "Improving target assessment in biomedical research: the GOT-IT recommendations". In: *Nature Reviews Drug Discovery* 20.1, pp. 64–81. DOI: 10.1038/s41573-020-0087-3 (cit. on p. 39).

Engvall, Eva and Peter Perlmann (1972). "Enzyme-Linked Immunosorbent Assay, Elisa". In: *The Journal of Immunology* 109.1, pp. 129–135 (cit. on p. 40).

Evans, Richard et al. (2021). "Protein complex prediction with AlphaFold-Multimer". In: *bioRxiv*. DOI: 10.1101/2021.10.04.463034 (cit. on pp. 24, 26, 61).

Falcon, William and Kyunghyun Cho (2020). "A Framework For Contrastive Self-Supervised Learning And Designing A New Approach". In: *arXiv*, arXiv:2009.00104 (cit. on p. 177).

Feige, Matthias J. et al. (2014). "The structural analysis of shark IgNAR antibodies reveals evolutionary principles of immunoglobulins". In: *Proceedings of the National Academy of*

Sciences of the United States of America 111.22, pp. 8155–8160. DOI: 10.1073/pnas.1321502111 (cit. on p. 6).

Ferdous, Saba and Andrew C R Martin (2018). "AbDb: antibody structure database-a database of PDB-derived antibody structures". In: *Database* 2018. DOI: 10.1093/database/bay040 (cit. on pp. 21, 78).

Fernández-Quintero, Monica L. et al. (2021). "Shark Antibody Variable Domains Rigidify Upon Affinity Maturation-Understanding the Potential of Shark Immunoglobulins as Therapeutics". In: *Frontiers in Molecular Biosciences* 8, p. 226. DOI: 10.3389/fmolb.2021.639166 (cit. on p. 6).

Fey, Matthias and Jan Eric Lenssen (2019). "Fast Graph Representation Learning with PyTorch Geometric". In: *arXiv*, arXiv:1903.02428 (cit. on p. 167).

Finzi, Marc et al. (2020). "Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data". In: *arXiv*, arXiv:2002.12880 (cit. on p. 55).

Foote, Jefferson and Greg Winter (1992). "Antibody framework residues affecting the conformation of the hypervariable loops". In: *Journal of Molecular Biology* 224.2, pp. 487–499. DOI: 10.1016/0022-2836(92)91010-M (cit. on p. 6).

Fuchs, Fabian B. et al. (2020a). "SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks". In: *arXiv*, arXiv:2006.10503 (cit. on pp. 54, 55, 162, 163, 171).

– (2020b). *SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks*. URL: https://github.com/FabianFuchsML/se3-transformer-public (visited on 12/10/2021) (cit. on p. 171).

Gainza, P. et al. (2020). "Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning". In: *Nature Methods* 17.2, pp. 184–192. DOI: 10.1038/s41592-019-0666-6 (cit. on pp. 58, 162).

Galson, Jacob D. et al. (2020). "Deep Sequencing of B Cell Receptor Repertoires From COVID-19 Patients Reveals Strong Convergent Immune Signatures". In: *Frontiers in Immunology* 11, p. 3283. DOI: 10.3389/fimmu.2020.605170 (cit. on p. 130).

Ganea, Octavian-Eugen et al. (2021). "Independent SE(3)-Equivariant Models for End-to-End Rigid Protein Docking". In: *arXiv*, arXiv:2111.07786 (cit. on pp. 61, 174, 197, 208).

Gao, C. et al. (2002). "A method for the generation of combinatorial antibody libraries using pIX phage display". In: *Proceedings of the National Academy of Sciences* 99.20, pp. 12612–12616. DOI: 10.1073/pnas.192467999 (cit. on p. 39).

Garbuzynskiy, Sergiy O. et al. (2005). "Comparison of X-ray and NMR structures: Is there a systematic difference in residue contacts between X-ray- and NMR-resolved protein structures?" In: *Proteins: Structure, Function, and Bioinformatics* 60.1, pp. 139–147. DOI: 10.1002/prot.20491 (cit. on p. 20).

Gaulton, A. et al. (2012). "ChEMBL: a large-scale bioactivity database for drug discovery". In: *Nucleic Acids Research* 40.D1, pp. D1100–D1107. DOI: 10.1093/nar/gkr777 (cit. on p. 57).

Ghani, Usman et al. (2021). "Improved Docking of Protein Models by a Combination of Alphafold2 and ClusPro". In: *bioRxiv*. DOI: 10.1101/2021.09.07.459290 (cit. on p. 61).

Giachino, C, E Padovan, and A Lanzavecchia (1995). "$\kappa+\lambda+$ dual receptor B cells are present in the human peripheral repertoire." In: *Journal of Experimental Medicine* 181.3, pp. 1245–1250. DOI: 10.1084/jem.181.3.1245 (cit. on p. 10).

Gilmer, Justin et al. (2017). "Neural message passing for quantum chemistry". In: *34th International Conference on Machine Learning, ICML 2017*. Vol. 3, pp. 2053–2070 (cit. on p. 52).

Giudicelli, V. (2004). "IMGT/GENE-DB: a comprehensive database for human and mouse immunoglobulin and T cell receptor genes". In: *Nucleic Acids Research* 33.Database issue, pp. D256–D261. DOI: 10.1093/nar/gki010 (cit. on pp. 8, 10).

Giudicelli, V. and M.-P. Lefranc (1999). "Ontology for immunogenetics: the IMGT-ONTOLOGY". In: *Bioinformatics* 15.12, pp. 1047–1054. DOI: 10.1093/bioinformatics/15.12.1047 (cit. on p. 22).

Giudicelli, Véronique, Denys Chaume, and Marie Paule Lefranc (2004). "IMGT/GENE-DB: a comprehensive database for human and mouse immunoglobulin and T cell receptor genes". In: *Nucleic Acids Research* 33.Database issue, pp. D256–D261. DOI: 10.1093/nar/gki010 (cit. on p. 22).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. MIT Press (cit. on pp. 46, 48).

Graves, Jordan et al. (2020). "A Review of Deep Learning Methods for Antibodies". In: *Antibodies* 9.2, p. 12. DOI: 10.3390/antib9020012 (cit. on p. 56).

Gray, A.C. et al. (2016). "Animal-Friendly Affinity Reagents: Replacing the Needless in the Haystack". In: *Trends in Biotechnology* 34.12, pp. 960–969. DOI: 10.1016/j.tibtech.2016.05.017 (cit. on p. 40).

Grilo, António L and A Mantalaris (2019). "The Increasingly Human and Profitable Monoclonal Antibody Market." In: *Trends in biotechnology* 37.1, pp. 9–16. DOI: 10.1016/j.tibtech.2018.05.014 (cit. on p. 14).

Gu, Kangxia et al. (2008). "Testing the ratio of two poisson rates". In: *Biometrical Journal* 50.2, pp. 283–298. DOI: 10.1002/bimj.200710403 (cit. on p. 98).

Guest, Johnathan D. et al. (2021). "An expanded benchmark for antibody-antigen docking and affinity prediction reveals insights into antibody recognition determinants". In: *Structure*. DOI: 10.1016/j.str.2021.01.005 (cit. on pp. 35, 36).

Gutzeit, Cindy, Kang Chen, and Andrea Cerutti (2018). "The enigmatic function of IgD: some answers at last". In: *European Journal of Immunology* 48.7, pp. 1101–1113. DOI: 10.1002/eji.201646547 (cit. on p. 2).

Hamers-Casterman, C. et al. (1993). "Naturally occurring antibodies devoid of light chains". In: *Nature* 363.6428, pp. 446–448. DOI: 10.1038/363446a0 (cit. on p. 17).

He, Kaiming et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 770–778. DOI: 10.1109/CVPR.2016.90 (cit. on p. 51).

Hearty, Stephen, Paul Leonard, and Richard O'Kennedy (2012). "Measuring antibody-antigen binding kinetics using surface plasmon resonance." In: *Methods in molecular biology* 907, pp. 411–42. DOI: 10.1007/978-1-61779-974-7_24 (cit. on p. 40).

Holliger, Philipp and Peter J Hudson (2005). "Engineered antibody fragments and the rise of single domains". In: *Nature Biotechnology* 23.9, pp. 1126–1136. DOI: 10.1038/nbt1142 (cit. on pp. 17, 68).

Hou, Qingzhen et al. (2021). "SeRenDIP-CE: sequence-based interface prediction for conformational epitopes". In: *Bioinformatics* 37.20, pp. 3421–3427. DOI: 10.1093/bioinformatics/btab321 (cit. on pp. 22, 67, 206).

Huang, Gao, Zhuang Liu, and Kilian Q. Weinberger (2016). "Densely Connected Convolutional Networks". In: *arXiv*, arXiv:1608.06993 (cit. on pp. 51, 125).

Huo, Jiangdong et al. (2020). "Neutralizing nanobodies bind SARS-CoV-2 spike RBD and block interaction with ACE2". In: *Nature Structural and Molecular Biology* 27.9, pp. 846–854. DOI: 10.1038/s41594-020-0469-6 (cit. on pp. 17, 68).

Hutchinson, Michael et al. (2021). "LieTransformer: Equivariant self-attention for Lie Groups". In: *arXiv*, arXiv:2012.10885 (cit. on pp. 54, 55, 162, 163).

Imrie, Fergus et al. (2018). "Protein Family-Specific Models Using Deep Neural Networks and Transfer Learning Improve Virtual Screening and Highlight the Need for More Data". In: *Journal of Chemical Information and Modeling* 58.11, pp. 2319–2330. DOI: 10.1021/acs.jcim.8b00350 (cit. on pp. 51, 56, 124, 202).

Imrie, Fergus et al. (2020). "Deep Generative Models for 3D Linker Design". In: *Journal of Chemical Information and Modeling* 60.4, pp. 1983–1995. DOI: 10.1021/acs.jcim.9b01120 (cit. on p. 52).

Jeliazkov, Jeliazko R et al. (2020). "Robustification of RosettaAntibody and Rosetta SnugDock". In: *bioRxiv*. DOI: https://doi.org/10.1101/2020.05.26.116210 (cit. on p. 35).

Jespersen, Martin Closter et al. (2019). "Antibody Specific B-Cell Epitope Predictions: Leveraging Information From Antibody-Antigen Protein Complexes." In: *Frontiers in immunology* 10, p. 298. DOI: 10.3389/fimmu.2019.00298 (cit. on p. 62).

Jiménez, J et al. (2017). "DeepSite: Protein-binding site predictor using 3D-convolutional neural networks". In: *Bioinformatics* 33.19, pp. 3036–3042. DOI: 10.1093/bioinformatics/btx350 (cit. on p. 57).

Jiménez, José et al. (2018). "KDEEP: Protein-Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks". In: *Journal of Chemical Information and Modeling* 58.2, pp. 287–296. DOI: 10.1021/acs.jcim.7b00650 (cit. on p. 57).

Jing, Bowen et al. (2020). "Learning from Protein Structure with Geometric Vector Perceptrons". In: *arXiv*, arXiv:2009.01411 (cit. on pp. 55, 162, 163, 189, 202).

Jing, Bowen et al. (2021). "Equivariant Graph Neural Networks for 3D Macromolecular Structure". In: *arXiv*, arXiv:2106.03843 (cit. on p. 55).

Jovčevska, Ivana and Serge Muyldermans (2020). "The Therapeutic Potential of Nanobodies". In: *BioDrugs* 34.1, pp. 11–26. DOI: 10.1007/s40259-019-00392-z (cit. on pp. 17, 68).

Jumper, John et al. (2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596.7873, pp. 583–589. DOI: 10.1038/s41586-021-03819-2 (cit. on pp. 24–26, 48, 52, 54, 55, 83, 85, 161, 162, 188, 202).

Kabat, Elvin Abraham, Tai Te Wu, and Howard Bilofsky (1979). *Sequences of immunoglobulin chains: tabulation and analysis of amino acid sequences of precursors, V-regions, C-regions, J-chain and 2-microglobulins*. English. National Institute of Health (cit. on p. 12).

Kang, Yue et al. (2021). "Sequence-based deep learning antibody design for in silico antibody affinity maturation". In: *arXiv*, arXiv:2103.03724 (cit. on pp. 44, 58).

Kelow, Simon P., Jared Adolf-Bryfogle, and Roland L. Dunbrack (2020). "Hiding in plain sight: structure and sequence analysis reveals the importance of the antibody DE loop for antibody-antigen binding". In: *mAbs* 12.1, p. 1840005. DOI: 10.1080/19420862.2020.1840005 (cit. on p. 7).

Kilambi, Krishna Praneeth and Jeffrey J. Gray (2017). "Structure-based cross-docking analysis of antibody-antigen interactions". In: *Scientific Reports* 7.1, p. 8145. DOI: 10.1038/s41598-017-08414-y (cit. on p. 158).

Kipf, Thomas N. and Max Welling (2017). "Semi-Supervised Classification with Graph Convolutional Networks". In: *arXiv*, arXiv:1609.02907 (cit. on pp. 163, 171, 182).

Klausen, Michael Schantz et al. (2015). "LYRA, a webserver for lymphocyte receptor structural modeling". In: *Nucleic Acids Research* 43.W1, W349–W355. DOI: 10.1093/nar/gkv535 (cit. on p. 29).

Kovaltsuk, Aleksandr et al. (2018). "Observed Antibody Space: A Resource for Data Mining Next-Generation Sequencing of Antibody Repertoires". In: *The Journal of Immunology* 201.8, pp. 2502–2509. DOI: 10.4049/jimmunol.1800708 (cit. on pp. 23, 40, 127, 131).

Kozakov, Dima et al. (2017). "The ClusPro web server for protein-protein docking". In: *Nature Protocols* 12.2, pp. 255–278. DOI: 10.1038/nprot.2016.169 (cit. on pp. 33, 35, 36).

Krah, Simon et al. (2016). "Single-domain antibodies for biomedical applications". In: *Immunopharmacology and Immunotoxicology* 38.1, pp. 21–28. DOI: 10.3109/08923973.2015.1102934 (cit. on p. 17).

Krawczyk, Konrad et al. (2013). "Antibody i-Patch prediction of the antibody binding site improves rigid local antibody-antigen docking". In: *Protein Engineering, Design and Selection* 26.10, pp. 621–629. DOI: 10.1093/protein/gzt043 (cit. on pp. 34, 43, 84, 90, 91, 106).

Krawczyk, Konrad et al. (2014). "Improving B-cell epitope prediction and its application to global antibody-antigen docking". In: *Bioinformatics* 30.16, pp. 2288–2294. DOI: 10.1093/bioinformatics/btu190 (cit. on p. 43).

Kringelum, Jens Vindahl et al. (2013). "Structural analysis of B-cell epitopes in antibody:protein complexes". In: *Molecular Immunology* 53 (1-2), pp. 24–34. DOI: 10.1016/j.molimm.2012.06.001 (cit. on p. 44).

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (2012). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems*. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc. (cit. on pp. 50, 51).

Krupa, Paweł et al. (2021). "UNRES-Dock—protein–protein and peptide–protein docking by coarse-grained replica-exchange MD simulations". In: *Bioinformatics* 37 (11), pp. 1613–1615. DOI: 10.1093/bioinformatics/btaa897 (cit. on p. 34).

Kryshtafovych, Andriy et al. (2019). "Critical assessment of methods of protein structure prediction (CASP)-Round XIII". In: *Proteins: Structure, Function, and Bioinformatics* 87.12, pp. 1011–1020. DOI: 10.1002/prot.25823 (cit. on p. 24).

– (2021). "Critical assessment of methods of protein structure prediction (CASP)-Round XIV". In: *Proteins: Structure, Function, and Bioinformatics* 89.12, pp. 1607–1617. DOI: 10.1002/prot.26237 (cit. on p. 24).

Kügler, Jonas et al. (2018). "Construction of Human Immune and Naive scFv Libraries". In: *Methods in Molecular Biology* 1701, pp. 3–24. DOI: 10.1007/978-1-4939-7447-4_1 (cit. on p. 42).

Kuhlman, Brian and Philip Bradley (2019). "Advances in protein structure prediction and design". In: *Nature Reviews Molecular Cell Biology* 20.11, pp. 681–697. DOI: 10.1038/s41580-019-0163-x (cit. on p. 23).

Kunik, Vered and Yanay Ofran (2013). "The indistinguishability of epitopes from protein surface is explained by the distinct binding preferences of each of the six antigen-binding loops". In: *Protein Engineering, Design and Selection* 26.10, pp. 599–609. DOI: 10.1093/protein/gzt027 (cit. on pp. 43, 44).

Labrijn, Aran F. et al. (2019). "Bispecific antibodies: a mechanistic review of the pipeline". In: *Nature Reviews Drug Discovery* 18.8, pp. 585–608. DOI: 10.1038/s41573-019-0028-1 (cit. on p. 18).

Lapidoth, Gideon D. et al. (2015). "AbDesign : An algorithm for combinatorial backbone design guided by natural conformations and sequences". In: *Proteins: Structure, Function, and Bioinformatics* 83.8, pp. 1385–1406. DOI: 10.1002/prot.24779 (cit. on pp. 44, 119).

Laustsen, Andreas H. et al. (2021). "Animal Immunization, in Vitro Display Technologies, and Machine Learning for Antibody Discovery". In: *Trends in Biotechnology* 39.12, pp. 1263–1273. DOI: 10.1016/j.tibtech.2021.03.003 (cit. on pp. 40–42).

Lazim, Raudah, Donghyuk Suh, and Sun Choi (2020). "Advances in Molecular Dynamics Simulations and Enhanced Sampling Methods for the Study of Protein Systems". In: *International Journal of Molecular Sciences* 21 (17), p. 6339. DOI: 10.3390/ijms21176339 (cit. on p. 34).

Lecun, Y. et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. DOI: 10.1109/5.726791 (cit. on pp. 50, 124).

LeCun, Yann, Yoshua Bengio, and Geoffrey E. Hinton (2015). "Deep learning". In: *Nature* 521.7553, pp. 436–444. DOI: 10.1038/nature14539 (cit. on pp. 46, 48, 50).

Lee, B. and F. M. Richards (1971). "The interpretation of protein structures: Estimation of static accessibility". In: *Journal of Molecular Biology* 55.3, 379–IN4. DOI: 10.1016/0022-2836(71)90324-X (cit. on p. 88).

Lee, E-Chiang et al. (2014). "Complete humanization of the mouse immunoglobulin loci enables efficient therapeutic antibody discovery". In: *Nature Biotechnology* 32.4, pp. 356–363. DOI: 10.1038/nbt.2825 (cit. on p. 42).

Leem, Jinwoo et al. (2016). "ABodyBuilder: Automated antibody structure prediction with data-driven accuracy estimation". In: *mAbs* 8.7, pp. 1259–1268. DOI: 10.1080/19420862.2016.1205773 (cit. on pp. 26, 27, 30, 83, 87, 100).

Leem, Jinwoo et al. (2018). "Antibody side chain conformations are position-dependent". In: *Proteins: Structure, Function and Bioinformatics* 86.4, pp. 383–392. DOI: 10.1002/prot.25453 (cit. on pp. 29, 86).

Lefranc, Marie-Paule et al. (2003). "IMGT unique numbering for immunoglobulin and T cell receptor variable domains and Ig superfamily V-like domains". In: *Developmental & Comparative Immunology* 27.1, pp. 55–77. DOI: 10.1016/S0145-305X(02)00039-3 (cit. on pp. 13, 85).

Lensink, Marc F. et al. (2020). "Modeling protein-protein, protein-peptide, and protein-oligosaccharide complexes: CAPRI 7th edition". In: *Proteins: Structure, Function, and Bioinformatics* 88.8, pp. 916–938. DOI: 10.1002/prot.25870 (cit. on p. 37).

Lensink, Marc F. et al. (2021). "Prediction of protein assemblies, the next frontier: The CASP14-CAPRI experiment". In: *Proteins: Structure, Function, and Bioinformatics* 89.12, pp. 1800–1823. DOI: 10.1002/prot.26222 (cit. on p. 37).

Lepore, Rosalba et al. (2017). "PIGSPro: prediction of immunoGlobulin structures v2". In: *Nucleic Acids Research* 45.W1, W17–W23. DOI: 10.1093/nar/gkx334 (cit. on p. 29).

Li, Lei et al. (2019). "AbRSA: A robust tool for antibody numbering". In: *Protein Science* 28.8, pp. 1524–1531. DOI: 10.1002/pro.3633 (cit. on p. 14).

Li, Tong et al. (2015). "Rigidity Emerges during Antibody Evolution in Three Distinct Antibody Systems: Evidence from QSFR Analysis of Fab Fragments". In: *PLOS Computational Biology* 11 (7), e1004327. DOI: 10.1371/journal.pcbi.1004327 (cit. on p. 36).

Li, Weizhong and Adam Godzik (2006). "Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences". In: *Bioinformatics* 22.13, pp. 1658–1659. DOI: 10.1093/bioinformatics/btl158 (cit. on p. 91).

Liberis, Edgar et al. (2018). "Parapred: antibody paratope prediction using convolutional and recurrent neural networks". In: *Bioinformatics* 34.17, pp. 2944–2950. DOI: 10.1093/bioinformatics/bty305 (cit. on pp. 43, 58, 89, 119).

Liu, Ge et al. (2019). "Antibody Complementarity Determining Region Design Using High-Capacity Machine Learning". In: *bioRxiv*. DOI: 10.1101/682880 (cit. on p. 44).

Liu, Justin K H (2014). "The history of monoclonal antibody development - Progress, remaining challenges and future innovations". In: *Annals of Medicine and Surgery* 3.4, pp. 113–116. DOI: 10.1016/j.amsu.2014.09.001 (cit. on p. 6).

Lo, Yu-Chen et al. (2018). "Machine learning in chemoinformatics and drug discovery". In: *Drug Discovery Today* 23.8, pp. 1538–1546. DOI: 10.1016/j.drudis.2018.05.010 (cit. on p. 56).

Lu, Ruei-Min et al. (2020). "Development of therapeutic antibodies for the treatment of diseases". In: *Journal of Biomedical Science* 27.1, p. 1. DOI: 10.1186/s12929-019-0592-z (cit. on p. 15).

Lyumkis, Dmitry (2019). "Challenges and opportunities in cryo-EM single-particle analysis". In: *Journal of Biological Chemistry* 294 (13), pp. 5181–5197. DOI: 10.1074/jbc.REV118.005602 (cit. on p. 20).

Ma, Jiabing et al. (2021). "Bispecific Antibodies: From Research to Clinical Application". In: *Frontiers in Immunology* 12. DOI: 10.3389/fimmu.2021.626616 (cit. on p. 18).

Maclin, R. and D. Opitz (1999). "Popular Ensemble Methods: An Empirical Study". In: *Journal of Artificial Intelligence Research* 11, pp. 169–198. DOI: 10.1613/jair.614 (cit. on pp. 140, 159).

Maia, Eduardo Habib Bechelane et al. (2020). "Structure-Based Virtual Screening: From Classical to Artificial Intelligence". In: *Frontiers in Chemistry* 8. DOI: 10.3389/fchem.2020.00343 (cit. on p. 45).

Mansoor, Sanaa et al. (2021). "Toward More General Embeddings for Protein Design: Harnessing Joint Representations of Sequence and Structure". In: *bioRxiv*. DOI: 10.1101/2021.09.01.458592 (cit. on p. 208).

Marks, Claire and Charlotte M. Deane (2020). "How repertoire data are changing antibody science". In: *Journal of Biological Chemistry* 295.29, pp. 9823–9837. DOI: 10.1074/jbc.REV120.010181 (cit. on p. 40).

Marks, Claire et al. (2017). "Sphinx: Merging knowledge-based and ab initio approaches to improve protein loop prediction". In: *Bioinformatics* 33.9, pp. 1346–1353. DOI: 10.1093/bioinformatics/btw823 (cit. on pp. 29, 100, 115).

Marks, Claire et al. (2021). "Humanization of antibodies using a machine learning approach on large-scale repertoire data". In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btab434 (cit. on pp. 40, 45).

Mason, Derek M. et al. (2021). "Optimization of therapeutic antibodies by predicting antigen specificity from antibody sequence via deep learning". In: *Nature Biomedical Engineering*, pp. 1–13. DOI: 10.1038/s41551-021-00699-9 (cit. on pp. 44, 45, 119).

McCallum, Matthew et al. (2021). "N-terminal domain antigenic mapping reveals a site of vulnerability for SARS-CoV-2". In: *Cell* 184.9, 2332–2347.e16. DOI: 10.1016/J.CELL.2021.03.028 (cit. on p. 246).

McQuin, Claire et al. (2018). "CellProfiler 3.0: Next-generation image processing for biology". In: *PLOS Biology* 16.7, e2005970. DOI: 10.1371/journal.pbio.2005970 (cit. on p. 51).

Mertens, Haydyn D.T. and Dmitri I. Svergun (2010). "Structural characterization of proteins and complexes using small-angle X-ray solution scattering". In: *Journal of Structural Biology* 172.1, pp. 128–141. DOI: 10.1016/j.jsb.2010.06.012 (cit. on p. 20).

Milesi, Alexandre (2021). *Accelerating SE(3)-Transformers Training Using an NVIDIA Open-Source Model Implementation*. URL: https://developer.nvidia.com/blog/accelerating-se3-transformers-training-using-an-nvidia-open-source-model-implementation/ (visited on 12/10/2021) (cit. on p. 177).

Mitchell, Tom (1997). *Machine Learning*. McGraw Hill (cit. on p. 46).

Morehead, Alex et al. (2021). "DIPS-Plus: The Enhanced Database of Interacting Protein Structures for Interface Prediction". In: *arXiv*, arXiv:2106.04362 (cit. on pp. 57, 207).

Moult, John et al. (1995). "A large-scale experiment to assess protein structure prediction methods". In: *Proteins: Structure, Function, and Genetics* 23.3, pp. ii–iv. DOI: 10.1002/prot.340230303 (cit. on p. 24).

Mullard, Asher (2021). "FDA approves 100th monoclonal antibody product". In: *Nature reviews drug discovery* 20.7, pp. 491–495. DOI: 10.1038/d41573-021-00079-7 (cit. on p. 14).

Murphy, Kenneth and Casey Weaver (2017). *Janeway's immunobiology [electronic resource]*. eng. 9th edition. New York, NY (cit. on pp. 2–4, 6, 10, 11).

Mysinger, Michael M. et al. (2012). "Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking". In: *Journal of Medicinal Chemistry* 55.14, pp. 6582–6594. DOI: 10.1021/jm300687e (cit. on p. 57).

Myung, Yoochan, Douglas E V Pires, and David B Ascher (2021). "CSM-AB: graph-based antibody-antigen binding affinity prediction and docking scoring function". In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btab762 (cit. on pp. 22, 63, 67).

Nair, Vinod and Geoffrey E. Hinton (2010). "Rectified linear units improve Restricted Boltzmann machines". In: *ICML 2010 - Proceedings, 27th International Conference on Machine Learning*, pp. 807–814 (cit. on p. 50).

Nakane, Takanori et al. (2020). "Single-particle cryo-EM at atomic resolution". In: *Nature* 587.7832, pp. 152–156. DOI: 10.1038/s41586-020-2829-0 (cit. on p. 20).

Narciso, Jo Erika T. et al. (2011). "Analysis of the antibody structure based on high-resolution crystallographic studies". In: *New Biotechnology* 28.5, pp. 435–447. DOI: 10.1016/j.nbt.2011.03.012 (cit. on pp. 4, 7, 8).

Norman, Richard A. et al. (2020). "Computational approaches to therapeutic antibody design: Established methods and emerging trends". In: *Briefings in Bioinformatics* 21.5, pp. 1549–1567. DOI: 10.1093/bib/bbz095 (cit. on pp. 27, 30, 33, 119).

North, Benjamin, Andreas Lehmann, and Roland L. Dunbrack (2011). "A New Clustering of Antibody CDR Loop Conformations". In: *Journal of Molecular Biology* 406.2, pp. 228–256. DOI: 10.1016/j.jmb.2010.10.030 (cit. on pp. 7, 13).

Olimpieri, Pier Paolo et al. (2013). "Prediction of site-specific interactions in antibody-antigen complexes: The proABC method and server". In: *Bioinformatics* 29.18, pp. 2285–2291. DOI: 10.1093/bioinformatics/btt369 (cit. on p. 119).

Pagadala, Nataraj S., Khajamohiddin Syed, and Jack Tuszynski (2017). "Software for molecular docking: a review". In: *Biophysical Reviews* 9.2, pp. 91–102. DOI: 10.1007/s12551-016-0247-1 (cit. on p. 31).

Pak, Myeongsuk and Sanghoon Kim (2017). "A review of deep learning in image recognition". In: *2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT)*. IEEE, pp. 1–3. DOI: 10.1109/CAIPT.2017.8320684 (cit. on p. 48).

Pearce, Robin and Yang Zhang (2021). "Toward the solution of the protein structure prediction problem". In: *Journal of Biological Chemistry* 297.1, p. 100870. DOI: 10.1016/j.jbc.2021.100870 (cit. on pp. 24, 25).

Pierce, Brian G., Yuichiro Hourai, and Zhiping Weng (2011). "Accelerating protein docking in ZDOCK using an advanced 3D convolution library". In: *PLoS ONE* 6.9, e24657. DOI: 10.1371/journal.pone.0024657 (cit. on pp. 33–35, 84, 87).

Pittala, Srivamshi and Chris Bailey-Kellogg (2020). "Learning context-aware structural representations to predict antigen and antibody binding interfaces". In: *Bioinformatics* 36.13. Ed. by Arne Elofsson, pp. 3996–4003. DOI: 10.1093/bioinformatics/btaa263 (cit. on pp. 43, 44, 59, 120, 162, 206).

Porter, Kathryn A et al. (2019). "What method to use for protein-protein docking?" In: *Current Opinion in Structural Biology* 55, pp. 1–7. DOI: 10.1016/j.sbi.2018.12.010 (cit. on pp. 33, 84).

Ragoza, Matthew et al. (2017). "Protein-Ligand Scoring with Convolutional Neural Networks". In: *Journal of Chemical Information and Modeling* 57.4, pp. 942–957. DOI: 10.1021/acs.jcim.6b00740 (cit. on pp. 56, 91, 96, 97, 124, 163).

Ramakrishnan, Raghunathan et al. (2014). "Quantum chemistry structures and properties of 134 kilo molecules". In: *Scientific Data* 1.1, p. 140022. DOI: 10.1038/sdata.2014.22 (cit. on p. 170).

Raybould, Matthew I. J. (2020). "Structure-aware tools for the development of therapeutic antibodies from natural immunoglobulins". PhD thesis. University of Oxford (cit. on pp. 45, 47).

Raybould, Matthew I. J., Wing Ki Wong, and Charlotte M. Deane (2019a). "Antibody-antigen complex modelling in the era of immunoglobulin repertoire sequencing". In: *Molecular Systems Design and Engineering* 4.4, pp. 679–688. DOI: 10.1039/c9me00034h (cit. on pp. 68, 119).

Raybould, Matthew I. J. et al. (2019b). "Five computational developability guidelines for therapeutic antibody profiling". In: *Proceedings of the National Academy of Sciences of the United States of America* 116.10, pp. 4025–4030. DOI: 10.1073/pnas.1810576116 (cit. on pp. 40, 45, 67, 88, 118, 121).

Raybould, Matthew I. J. et al. (2020). "Thera-SAbDab: the Therapeutic Structural Antibody Database". In: *Nucleic Acids Research* 48.D1, pp. D383–D388. DOI: 10.1093/nar/gkz827 (cit. on pp. 17, 18, 22, 67).

Raybould, Matthew I. J. et al. (2021a). "CoV-AbDab: the coronavirus antibody database". In: *Bioinformatics* 37.5, pp. 734–735. DOI: 10.1093/bioinformatics/btaa739 (cit. on pp. 22, 67, 131).

Raybould, Matthew I. J. et al. (2021b). "Public Baseline and shared response structures support the theory of antibody repertoire functional commonality". In: *PLOS Computational Biology* 17.3, e1008781. DOI: 10.1371/journal.pcbi.1008781 (cit. on pp. 40, 45, 154).

Réau, Manon et al. (2021). "DeepRank-GNN: A Graph Neural Network Framework to Learn Patterns in Protein-Protein Interfaces". In: *bioRxiv*. DOI: 10.1101/2021.12.08.471762 (cit. on pp. 60, 204, 206).

Rees, Anthony R. (2020). "Understanding the human antibody repertoire". In: *mAbs* 12.1, p. 1729683. DOI: 10.1080/19420862.2020.1729683 (cit. on pp. 3, 23).

Renaud, Nicolas et al. (2021). "DeepRank: a deep learning framework for data mining 3D protein-protein interfaces". In: *Nature Communications 2021 12:1* 12.1, pp. 1–8. DOI: 10.1038/s41467-021-27396-0 (cit. on p. 60).

Ripoll, Daniel R., Sidhartha Chaudhury, and Anders Wallqvist (2021). "Using the antibody-antigen binding interface to train image-based deep neural networks for antibody-epitope classification". In: *PLOS Computational Biology* 17.3, e1008864. DOI: 10.1371/journal.pcbi.1008864 (cit. on p. 62).

Robert, Philippe A. et al. (2021). "One billion synthetic 3D-antibody-antigen complexes enable unconstrained machine-learning formalized investigation of antibody specificity prediction". In: *bioRxiv*. DOI: 10.1101/2021.07.06.451258 (cit. on pp. 62, 207).

Robinson, Sarah A et al. (2021). "Epitope profiling of coronavirus-binding antibodies using computational structural modelling". In: *bioRxiv*. DOI: 10.1101/2021.04.12.439478 (cit. on pp. 22, 44, 67).

Roth, David B. (2014). "V(D)J Recombination: Mechanism, Errors, and Fidelity". In: *Microbiology Spectrum* 2.6. DOI: 10.1128/microbiolspec.MDNA3-0041-2014 (cit. on pp. 8, 10).

Ruffolo, Jeffrey A, Jeremias Sulam, and Jeffrey J Gray (2021). "Antibody structure prediction using interpretable deep learning". In: *bioRxiv*. DOI: 10.1101/2021.05.27.445982 (cit. on pp. 26, 30).

Ruffolo, Jeffrey A et al. (2020). "Geometric potentials from deep learning improve prediction of CDR H3 loop structures". In: *Bioinformatics* 36.Supplement_1, pp. i268–i275. DOI: 10.1093/bioinformatics/btaa457 (cit. on pp. 26, 30, 115).

Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams (1986). "Learning representations by back-propagating errors". In: *Nature* 323.6088, pp. 533–536. DOI: 10.1038/323533a0 (cit. on p. 48).

Safdari, Yaghoub et al. (2013). "Antibody humanization methods - a review and update". In: *Biotechnology and Genetic Engineering Reviews* 29.2, pp. 175–186. DOI: 10.1080/02648725.2013.801235 (cit. on p. 42).

Saka, Koichiro et al. (2021). "Antibody design using LSTM based deep generative model from phage display library for affinity maturation". In: *Scientific Reports* 11.1, p. 5852. DOI: 10.1038/s41598-021-85274-7 (cit. on p. 58).

Šali, Andrej and Tom L. Blundell (1993). "Comparative Protein Modelling by Satisfaction of Spatial Restraints". In: *Journal of Molecular Biology* 234.3, pp. 779–815. DOI: 10.1006/jmbi.1993.1626 (cit. on p. 29).

Samsudin, Firdaus et al. (2020). "Not all therapeutic antibody isotypes are equal: the case of IgM versus IgG in Pertuzumab and Trastuzumab". In: *Chemical Science* 11.10, pp. 2843–2854. DOI: 10.1039/C9SC04722K (cit. on p. 16).

Satorras, Victor Garcia, Emiel Hoogeboom, and Max Welling (2021). "E(n) Equivariant Graph Neural Networks". In: *arXiv*, arXiv:2102.09844 (cit. on pp. 52–55, 61, 162, 163, 174, 175, 182, 184, 188, 200).

Scantlebury, Jack et al. (2020). "Data Set Augmentation Allows Deep Learning-Based Virtual Screening to Better Generalize to Unseen Target Classes and Highlight Important Binding Interactions". In: *Journal of Chemical Information and Modeling* 60.8, pp. 3722–3730. DOI: 10.1021/acs.jcim.0c00263 (cit. on pp. 56, 125, 136).

Scarselli, F. et al. (2009). "The Graph Neural Network Model". In: *IEEE Transactions on Neural Networks* 20.1, pp. 61–80. DOI: 10.1109/TNN.2008.2005605 (cit. on p. 52).

Scheid, Johannes F. et al. (2021). “B cell genomics behind cross-neutralization of SARS-CoV-2 variants and SARS-CoV”. In: *Cell* 184.12, 3205–3221.e24. DOI: 10.1016/J.CELL.2021.04.032 (cit. on p. 246).

Schlander, Michael et al. (2021). “How Much Does It Cost to Research and Develop a New Drug? A Systematic Review and Assessment”. In: *PharmacoEconomics* 39.11, pp. 1243–1269. DOI: 10.1007/s40273-021-01065-y (cit. on p. 39).

Schneider, Constantin, Matthew I. J. Raybould, and Charlotte M. Deane (2021a). “SAbDab in the age of biotherapeutics: updates including SAbDab-nano, the nanobody structure tracker”. In: *Nucleic Acids Research*. gkab1050. DOI: 10.1093/nar/gkab1050 (cit. on pp. 66, 67).

Schneider, Constantin et al. (2021b). “DLAB: deep learning methods for structure-based virtual screening of antibodies”. In: *Bioinformatics*, btab660. DOI: 10.1093/bioinformatics/btab660 (cit. on pp. 22, 67, 82, 118, 163).

Schritt, Dimitri et al. (2019). “Repertoire Builder: high-throughput structural modeling of B and T cell receptors”. In: *Molecular Systems Design & Engineering* 4.4, pp. 761–768. DOI: 10.1039/C9ME00020H (cit. on p. 29).

Schymkowitz, Joost et al. (2005). “The FoldX web server: an online force field”. In: *Nucleic Acids Research* 33.suppl_2, W382–W388. DOI: 10.1093/nar/gki387 (cit. on p. 131).

Seabold, Skipper and Josef Perktold (2010). “statsmodels: Econometric and statistical modeling with python”. In: *9th Python in Science Conference* (cit. on p. 99).

Simonyan, Karen and Andrew Zisserman (2015). “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *International Conference on Learning Representations* (cit. on p. 51).

Sircar, Aroop and Jeffrey J. Gray (2010). “SnugDock: Paratope Structural Optimization during Antibody-Antigen Docking Compensates for Errors in Antibody Homology Models”. In: *PLoS Computational Biology* 6.1, e1000644. DOI: 10.1371/journal.pcbi.1000644 (cit. on pp. 33, 35).

Slabodkin, Andrei et al. (2021). “Individualized VDJ recombination predisposes the available Ig sequence space”. In: *Genome Research*. DOI: 10.1101/gr.275373.121 (cit. on p. 10).

Sormanni, Pietro, Francesco A. Aprile, and Michele Vendruscolo (2018). “Third generation antibody discovery methods: In silico rational design”. In: *Chemical Society Reviews* 47.24, pp. 9137–9157. DOI: 10.1039/c8cs00523k (cit. on pp. 40, 42, 44).

Spiess, Christoph, Qianting Zhai, and Paul J. Carter (2015). “Alternative molecular formats and therapeutic applications for bispecific antibodies”. In: *Molecular Immunology* 67.2, pp. 95–106. DOI: 10.1016/j.molimm.2015.01.003 (cit. on p. 18).

Srivastava, Nitish et al. (2014). "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15, pp. 1929–1958. DOI: 10.1214/12-AOS1000 (cit. on p. 50).

Sterlin, Delphine and Guy Gorochov (2021). "When Therapeutic IgA Antibodies Might Come of Age". In: *Pharmacology* 106.1-2, pp. 9–19. DOI: 10.1159/000510251 (cit. on p. 16).

Storb, U. et al. (1999). "Molecular Aspects of Somatic Hypermutation of Immunoglobulin Genes". In: *Cold Spring Harbor Symposia on Quantitative Biology* 64, pp. 227–234. DOI: 10.1101/sqb.1999.64.227 (cit. on p. 11).

Sui, Jianhua et al. (2005). "Evaluation of Human Monoclonal Antibody 80R for Immunoprophylaxis of Severe Acute Respiratory Syndrome by an Animal Study, Epitope Mapping, and Analysis of Spike Variants". In: *Journal of Virology* 79.10, pp. 5900–5906. DOI: 10.1128/JVI.79.10.5900-5906.2005 (cit. on p. 130).

Sui, Jianhua et al. (2008). "Broadening of Neutralization Activity to Directly Block a Dominant Antibody-Driven SARS-Coronavirus Evolution Pathway". In: *PLOS Pathogens* 4.11, pp. 1–14. DOI: 10.1371/journal.ppat.1000197 (cit. on p. 130).

Sun, Zehua et al. (2021). "Neutralization of European, South African, and United States SARS-CoV-2 mutants by a human antibody and antibody domains". In: *bioRxiv*. DOI: 10.1101/2021.03.22.436481 (cit. on p. 246).

Sunseri, Jocelyn and David Ryan Koes (2019). "libmolgrid: GPU Accelerated Molecular Gridding for Deep Learning Applications". In: *Journal of Chemical Information and Modeling*, acs.jcim.9b01145. DOI: 10.1021/acs.jcim.9b01145 (cit. on pp. 56, 91).

Suscovich, Todd J and Galit Alter (2014). "In situ production of therapeutic monoclonal antibodies". In: *Expert Review of Vaccines* 14.2, pp. 205–219. DOI: 10.1586/14760584.2015.1001375 (cit. on p. 118).

Szegedy, Christian et al. (2014). "Going Deeper with Convolutions". In: *arXiv*, arXiv:1409.4842. DOI: 10.1109/CVPR.2015.7298594 (cit. on p. 51).

Teng, Yumin et al. (2020). "Diverse human VH antibody fragments with bio-therapeutic properties from the Crescendo Mouse". In: *New Biotechnology* 55, pp. 65–76. DOI: 10.1016/j.nbt.2019.10.003 (cit. on p. 42).

Thomas, Nathaniel et al. (2018). "Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds". In: *arXiv*, arXiv:1802.08219 (cit. on p. 55).

Tiller, Thomas et al. (2013). "A fully synthetic human Fab antibody library based on fixed VH/VL framework pairings with favorable biophysical properties". In: *mAbs* 5.3, pp. 445–470. DOI: 10.4161/mabs.24218 (cit. on p. 42).

Torchala, Mieczyslaw et al. (2013). "SwarmDock: a server for flexible protein-protein docking". In: *Bioinformatics* 29.6, pp. 807–809. DOI: 10.1093/bioinformatics/btt038 (cit. on pp. 35, 36).

Tortorici, M. Alejandra et al. (2021). "Structural basis for broad sarbecovirus neutralization by a human monoclonal antibody". In: *bioRxiv*. DOI: 10.1101/2021.04.07.438818 (cit. on p. 246).

Townshend, Raphael J L et al. (2020). "ATOM3D: Tasks On Molecules in Three Dimensions". In: *arXiv*, arXiv:2012.04035 (cit. on p. 57).

Townshend, Raphael J. L. et al. (2019). "End-to-end learning on 3D protein structure for interface prediction". In: *Advances in Neural Information Processing Systems*. Vol. 32 (cit. on pp. 57, 85, 159, 207).

Tunyasuvunakool, Kathryn et al. (2021). "Highly accurate protein structure prediction for the human proteome". In: *Nature* 596.7873, pp. 590–596. DOI: 10.1038/s41586-021-03828-1 (cit. on p. 25).

Turk, Martin and Wolfgang Baumeister (2020). "The promise and the challenges of cryo-electron tomography". In: *FEBS Letters* 594.20, pp. 3243–3261. DOI: 10.1002/1873-3468.13948 (cit. on p. 19).

U.S. Food and Drug Administration (2018). *The Drug Development Process - Step 3: Clinical Research*. https://www.fda.gov/patients/drug-development-process/step-3-clinical-research. Accessed: 2021-12-01 (cit. on p. 39).

Vamathevan, Jessica et al. (2019). "Applications of machine learning in drug discovery and development". In: *Nature Reviews Drug Discovery* 18.6, pp. 463–477. DOI: 10.1038/s41573-019-0024-5 (cit. on p. 56).

Vaswani, Ashish et al. (2017). "Attention Is All You Need". In: *Advances in Neural Information Processing Systems* (cit. on pp. 48, 55).

Virtanen, Pauli et al. (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17, pp. 261–272. DOI: 10.1038/s41592-019-0686-2 (cit. on p. 98).

Wan, Zihui, Yaofeng Zhao, and Yi Sun (2021). "Immunoglobulin D and its encoding genes: An updated review". In: *Developmental and Comparative Immunology* 124, p. 104198. DOI: 10.1016/j.dci.2021.104198 (cit. on pp. 2, 39).

Wang, Minjie et al. (2019a). "Deep Graph Library: Towards Efficient and Scalable Deep Learning on Graphs". In: *arXiv*, arXiv:1909.01315 (cit. on p. 171).

Wang, Xiao, Sean T. Flannery, and Daisuke Kihara (2021a). "Protein Docking Model Evaluation by Graph Neural Networks". In: *Frontiers in Molecular Biosciences* 8. DOI: 10.3389/fmolb.2021.647915 (cit. on pp. 60, 204, 206).

Wang, Xiao et al. (2019b). "Protein docking model evaluation by 3D deep convolutional neural networks". In: *Bioinformatics* 36.7, pp. 2113–2118. DOI: 10.1093/bioinformatics/btz870 (cit. on pp. 60, 84, 97, 113, 114).

– (2021b). *Kiharalab/dove: A deep-learning based docking decoy evaluation method*. URL: https://github.com/kiharalab/DOVE (visited on 10/05/2021) (cit. on p. 98).

Weiler, Maurice et al. (2018). "3D Steerable CNNs: Learning Rotationally Equivariant Features in Volumetric Data". In: *arXiv*, p. 1807.02547 (cit. on pp. 162, 163).

Weitzner, Brian D et al. (2017). "Modeling and docking of antibody structures with Rosetta". In: *Nature Protocols* 12.2, pp. 401–416. DOI: 10.1038/nprot.2016.180 (cit. on pp. 29, 33, 83, 84, 115).

Wilton, Emily E. et al. (2018). *SdAb-DB: The Single Domain Antibody Database*. DOI: 10.1021/acssynbio.8b00407 (cit. on p. 78).

Wodak, Shoshana J and Raúl Méndez (2004). "Prediction of protein-protein interactions: The CAPRI experiment, its evaluation and implications". In: *Current Opinion in Structural Biology* 14.2, pp. 242–249. DOI: 10.1016/j.sbi.2004.02.003 (cit. on pp. 37, 89, 114).

Wong, Wing Ki et al. (2019). "SCALOP: sequence-based antibody canonical loop structure annotation". In: *Bioinformatics* 35.10, pp. 1774–1776. DOI: 10.1093/bioinformatics/bty877 (cit. on p. 7).

Wouters, Olivier J., Martin McKee, and Jeroen Luyten (2020). "Estimated Research and Development Investment Needed to Bring a New Medicine to Market, 2009-2018". In: *JAMA* 323.9, p. 844. DOI: 10.1001/jama.2020.1166 (cit. on p. 39).

Wu, Zonghan et al. (2021). "A Comprehensive Survey on Graph Neural Networks". In: *IEEE Transactions on Neural Networks and Learning Systems* 32.1, pp. 4–24. DOI: 10.1109/TNNLS.2020.2978386 (cit. on pp. 52, 53).

Xenarios, I. (2002). "DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions". In: *Nucleic Acids Research* 30.1, pp. 303–305. DOI: 10.1093/nar/30.1.303 (cit. on p. 57).

Yamashita, Kazuo et al. (2014). "Kotai Antibody Builder: Automated high-resolution structural modeling of antibodies". In: *Bioinformatics* 30.22, pp. 3279–3280. DOI: 10.1093/bioinformatics/btu510 (cit. on p. 83).

Yang, Emily Y. and Khalid Shah (2020). "Nanobodies: Next Generation of Cancer Diagnostics and Therapeutics". In: *Frontiers in Oncology* 10, p. 1182. DOI: 10.3389/fonc.2020.01182 (cit. on pp. 17, 68).

Yasser, Mohseni Behbahani, Élodie Laine, and Alessandra Carbone (2021). "DLA-Ranker: Evaluating protein docking conformations with many locally oriented cubes". In: *bioRxiv*. DOI: 10.1101/2021.10.26.465898 (cit. on p. 60).

Yin, Rui et al. (2021). "Benchmarking AlphaFold for protein complex modeling reveals accuracy determinants". In: *bioRxiv*. DOI: 10.1101/2021.10.23.465575 (cit. on pp. 24, 61).

Zavrtanik, Uroš and San Hadži (2019). "A non-redundant data set of nanobody-antigen crystal structures". In: *Data in Brief* 24, p. 103754. DOI: 10.1016/j.dib.2019.103754 (cit. on p. 78).

Zenkova, Natalia et al. (2021). "Simple End-to-end Deep Learning Model for CDR-H3 Loop Structure Prediction". In: *arXiv*, arXiv:2111.10656 (cit. on pp. 26, 30).

Zhou, Jie et al. (2020). "Graph neural networks: A review of methods and applications". In: *AI Open* 1, pp. 57–81. DOI: 10.1016/j.aiopen.2021.01.001 (cit. on p. 53).

A | High-throughput generation and assessment of antibody-antigen complex poses

Contents

A.1 Atom types used during grid generation

Type	Notes
AliphaticCarbonXSHydrophobe	Hydrophobic carbon atoms not in an aromatic system
AliphaticCarbonXSNonHydrophobe	Hydrophilic carbon atoms not in an aromatic system
AromaticCarbonXSHydrophobe	Hydrophobic carbon atoms in an aromatic system
AromaticCarbonXSNonHydrophobe	Hydrophilic carbon atoms in an aromatic system
Halogens	Contains Bromine, Iodine, Chlorine, Fluorine
Nitrogen NitrogenXSAcceptor	Nitrogen atoms not able to act as electron donors
NitrogenXSDonor NitrogenXSDonorAcceptor	Nitrogen atoms able to act as electron donors
Oxygen OxygenXSAcceptor	Oxygen atoms not able to act as electron donors
OxygenXSDonorAcceptor OxygenXSDonor	Oxygen atoms able to act as electron donors
Sulphur & SulphurAcceptor	
Phosphorus	
Calcium	
Zinc	
Other	

Table A.1: Atom types used to generate atom density grids with libmolgrid.

A.2 PDB files used to generate the crystal structure and model data set

Table A.2: PDB accession codes used to generate the crystal structure and model data set.

1a2y	1uj3	2xzq	3mnz	4ers	4m5z	4xmp	5eor	5tkj	6bf4
1a3r	1uwx	2y5t	3mxw	4etq	4m62	4xnq	5epm	5tkk	6bit
1adq	1v7m	2y6s	3nfp	4f15	4m7l	4xny	5erw	5tpn	6blh
1afv	1w72	2yc1	3nh7	4f2m	4m8q	4xnz	5esv	5tq0	6bli
1ahw	1wej	2ypv	3o2d	4f37	4mhh	4xp4	5eu7	5tqq	6bp2
1ai1	1xgp	2zch	3o45	4f3f	4mwf	4xtr	5ezo	5tru	6bqb
1bj1	1xgq	2zjs	3o6l	4ffv	4mxv	4xvj	5f96	5tud	6c6y
1ce1	1xgr	2zuq	3opz	4ffy	4n0y	4xvt	5f9o	5u6a	6c6z
1cu4	1xgt	3a67	3p0y	4fp8	4n8c	4xzu	5f9w	5u7m	6elu
1cz8	1xgu	3a6b	3p30	4g3y	4n9g	4y5v	5fb8	5u8r	6eti
1e4x	1xgy	3a6c	3pgf	4g6j	4np4	4y5x	5fcu	5ucb	6fax
1egj	1xiw	3b2u	3pjs	4g6m	4nzr	4y5y	5fec	5udc	6fn1
1ejo	1ynt	3b9k	3pnw	4g7v	4nzt	4ybq	5fgb	5uea	
1f58	1yqv	3bdy	3pp4	4g80	4o02	4yc2	5fgc	5uem	
1f90	1yy9	3bgf	3qa3	4gag	4o4y	4ydi	5ggr	5ug0	
1fbi	1z3g	3bkj	3qg6	4gms	4o51	4ydj	5ggs	5ugy	
1fe8	1za3	3bky	3qnz	4gxu	4o58	4ydk	5ggt	5ukr	
1fj1	1zea	3bn9	3qum	4h88	4o9h	4ydl	5ggv	5umn	
1fns	1ztx	3bsz	3qwo	4h8w	4od2	4ydv	5gis	5ush	
1fpt	2a6i	3c09	3raj	4hc1	4odx	4ye4	5gjs	5usl	
1frg	2adf	3cvh	3rhw	4hcr	4ogy	4yhz	5gmq	5v6l	
1fsk	2aep	3cx5	3s35	4hf5	4oii	4yk4	5gzo	5v6m	
1g7j	2ap2	3cxd	3s37	4hfu	4ojf	4yo0	5h35	5v7j	
1g7l	2arj	3d85	3s88	4hg4	4okv	4yr6	5hdq	5vag	

1g7m	2b1h	3d9a	3sdy	4hha	4olu	4ywg	5hi4	5vcn
1g9m	2bdn	3dvg	3se9	4hj0	4olv	4yzf	5hj3	5vco
1ggi	2ck0	3dvn	3sge	4hkx	4olw	4z0x	5hys	5veb
1h0d	2dd8	3e8u	3skj	4hlz	4olx	4zfg	5i6x	5vgj
1hi6	2dqg	3efd	3sqo	4hpo	4olz	4zfo	5i8c	5vic
1i9r	2eh8	3eff	3t2n	4hpy	4om0	4zpt	5i9q	5vig
1ic5	2eiz	3eoa	3t3p	4hs6	4om1	4zs6	5ies	5vjo
1ifh	2fd6	3eyf	3tt1	4hs8	4oqt	4zs7	5ig7	5vjq
1ikf	2fjg	3ffd	3tt3	4ht1	4p3c	4zso	5igx	5vl3
1iqd	2fjh	3fmg	3u30	4hwb	4p59	4zto	5ijk	5vl7
1j1o	2g5b	3fn0	3u4e	4hzl	4plj	4zxb	5ikc	5vlp
1j1p	2gsi	3g04	3u9p	4i18	4ps4	5anm	5j3h	5vob
1j1x	2h1p	3g5v	3ubx	4i77	4py8	5aum	5jhl	5vpl
1jhl	2h9g	3g5y	3uc0	4idj	4q0x	5b3j	5jq6	5vxr
1jps	2hfg	3g6d	3uji	4irz	4q6i	5b71	5jw3	5vyf
1jrh	2hh0	3gbn	3ulu	4j4p	4qci	5bjz	5jz7	5w06
1kb5	2hrp	3ggw	3ulv	4j6r	4qhu	5bk1	5k59	5w08
1kc5	2i9l	3ghe	3v4u	4j8r	4qti	5bk2	5k9k	5w0d
1kcr	2igf	3gi9	3v4v	4jan	4qww	5bo1	5k9o	5w0k
1kcs	2ih3	3gjf	3v6o	4jb9	4qy8	5bv7	5kaq	5w23
1ken	2ipu	3go1	3v6z	4jdt	4r0l	5c0n	5kjr	5w3m
1kip	2j4w	3grw	3v7a	4jfz	4r2g	5c6t	5kn5	5w3p
1kiq	2j88	3h0t	3vga	4jg1	4r8w	5c7x	5kte	5w6d
1kir	2jel	3h3p	3vi4	4jhw	4rau	5c8j	5kvd	5w6g
1ktr	2jix	3h42	3vrl	4jkp	4rav	5cba	5kve	5w9i
1mhp	2ny1	3hae	3w2d	4jo1	4rdq	5cbe	5kvf	5wdf
1mlc	2nyy	3hb3	3w9e	4jo3	4rgm	5cd5	5kvg	5whk
1mpa	2nz9	3hi1	3wd5	4jpk	4rgo	5cez	5kw9	5wk3
1mvu	2oqj	3hi6	3wfd	4jpv	4ris	5cjo	5l6y	5wnb

1n0x	2otu	3hmx	3whe	4jpw	4rwy	5csz	5lbs	5wux
1n64	2oz4	3hr5	3wih	4jqi	4rx4	5d1q	5lcv	5x08
1nak	2q8b	3i50	3wkm	4jzj	4s1q	5d1x	5ldn	5x0t
1nby	2qad	3idg	3wsq	4jzo	4s1r	5d1z	5lqb	5xcs
1ncb	2qqk	3idx	3x3f	4k2u	4s1s	5d70	5lsp	5xct
1ndg	2qqn	3iet	3zkm	4k3j	4tqe	5d71	5lwy	5xhv
1ndm	2qr0	3ifl	3ztn	4k94	4tsa	5d72	5mes	5xku
1nfd	2qsc	3ifo	4aei	4k9e	4tsb	5d8j	5mu0	5xmh
1nmb	2r0k	3ifp	4al8	4ki5	4tsc	5d93	5mvz	5xs7
1nsn	2r0l	3iu3	4bh8	4kro	4ttd	5d96	5myk	5xwd
1ob1	2r0w	3k2u	4bz1	4krp	4tul	5dd0	5myx	5xxy
1orq	2r29	3kj4	4bz2	4kuc	4u0r	5dfv	5ngv	5y11
1ors	2r56	3kj6	4cad	4kxz	4u1g	5dlm	5nj6	5y9j
1osp	2uzi	3kr3	4cmh	4l5f	4u6h	5dmg	5nph	6aod
1p2c	2v17	3ks0	4cni	4leo	4u6v	5do2	5nuz	6apb
1p4b	2vis	3l5x	4d3c	4lf3	4ut6	5drz	5o14	6aq7
1pz5	2vwe	3l95	4dgy	4liq	4ut9	5dsc	5o6v	6b08
1qfu	2vxq	3ldb	4dke	4lkx	4uu9	5dum	5sx5	6b0a
1qfw	2vxt	3lex	4dkf	4lmq	4v1d	5dup	5sy8	6b0e
1qkz	2w65	3ley	4dn4	4lsp	4web	5dur	5t33	6b0g
1r0a	2wuc	3lqa	4dqo	4lst	4wfe	5dwu	5t3z	6b0h
1rjl	2x7l	3lzf	4dtg	4lsu	4wuu	5e8d	5t6p	6b0s
1rvf	2xqb	3ma9	4dvr	4lu5	4xak	5e94	5tbd	6b5m
1sm3	2xqy	3mac	4dw2	4lvh	4xcf	5ea0	5te7	6b5r
1sy6	2xra	3mj9	4edw	4lvn	4xgz	5en2	5tfw	6b70
1tqb	2xtj	3mlv	4edx	4m1d	4xh2	5eoc	5th9	6b9j
1uac	2xwt	3mly	4ene	4m1g	4xmk	5eoq	5tih	6bdz

B | Classification of antibody-antigen binding

Contents

B.1 PDB files used to generate the post-snapshot test set

6j11	6k5i	6njm	6oe5	6p65	6pz8	6qno	6u1n	6uyd	6yor
6j14	6k65	6njn	6oel	6p8m	6pze	6r0x	6u2f	6uye	7bu8
6j15	6k68	6nm6	6oge	6p8n	6pzf	6r2s	6u36	6uyf	7bua
6j5d	6k7o	6nmr	6ogx	6p91	6pzw	6r8x	6u38	6uyg	7bue
6j5f	6kn9	6nms	6oij	6p95	6pzy	6rcu	6u3i	6uym	7buf
6j5g	6kpf	6nmu	6oik	6pa0	6pzz	6rcv	6u59	6v4p	7bz5
6j6y	6kpg	6nn3	6okm	6pci	6q0e	6rps	6u6u	6v8x	
6j71	6l62	6nnf	6okn	6pcu	6q0h	6s3d	6u9s	6v8z	
6jbt	6lgw	6nnj	6okp	6pdx	6q0i	6s3t	6uda	6vel	
6jep	6lhp	6nqd	6olp	6pe8	6q0l	6s5a	6udj	6vgr	
6jfh	6lhq	6nyq	6omm	6pe9	6q0o	6s8d	6udk	6vja	
6jfi	6lht	6nz7	6oor	6phb	6q18	6s8i	6ujb	6vrh	
6jhq	6lml	6o1f	6orn	6phc	6q1z	6s8j	6ujc	6vrk	
6jhr	6ln2	6o39	6os9	6phd	6q20	6snh	6ukj	6vrl	
6jhs	6lnt	6o3a	6osh	6phf	6q23	6sni	6um5	6vy2	
6jht	6lz9	6o3b	6osv	6pi7	6qb3	6svl	6um6	6vzi	
6jjp	6m58	6o41	6osy	6pis	6qb4	6t3f	6um7	6w03	
6jod	6nfu	6o9h	6ot0	6plk	6qb6	6tou	6umg	6w2b	
6k0y	6nfv	6o9i	6ot1	6ppg	6qd7	6txz	6umx	6w2c	
6k41	6nha	6oan	6otc	6pt0	6qd8	6tyb	6uqr	6w41	
6k42	6nij	6oao	6ous	6pv7	6qee	6tys	6urh	6xxv	
6k5a	6nip	6oc3	6oy4	6pv8	6qex	6u02	6usf	6y9b	
6k5d	6niu	6ocb	6oz9	6pwc	6qfc	6u0l	6uta	6yla	
6k5f	6njl	6oe4	6p62	6pxh	6qig	6u0n	6ute	6ym0	

Table B.1: PDB accession codes used to generate the post-snapshot test set

B.2 Antibodies used to create the epitope cluster test sets

B.2.1 Lysozyme epitope cluster

Table B.2: The cognate antibodies used to create the lysozyme epitope cluster. CDRs are determined using the IMGT CDR definition.

PDB code	Chain	CDR1	CDR2	CDR3
1fbi	Heavy	GYTFTSYW	IDPSDSYP	ASLYYYGTSYGVLDY
	Light	QDISNY	YTS	QQGYTLPYT
1nby	Heavy	GDSVTSDY	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT
1ndg	Heavy	GDSIIRDY	ISFSGNT	ANWDGTY
	Light	QSISNN	YAS	QQSNSWPYT
1ndm	Heavy	GDSITSDY	ISYSGST	ARWEMDY
	Light	QSISNN	YAS	QQSNSWPYT
1xgp	Heavy	GDSVTSDA	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT
1xgq	Heavy	GDSVTSDV	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT
1xgr	Heavy	GDSVTSDI	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT
1xgt	Heavy	GDSVTSDL	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT
1xgu	Heavy	GDSVTSDF	ISYSGST	ASWGGDV
	Light	QSISNN	YAS	QQSNSWPYT

PDB code	Chain	CDR1	CDR2	CDR3
5vjq	Heavy	GDSFTSDY	VTFSGST	ANWDGDY
	Light	QSIGNN	YAS	QQSNSWPYT

B.2.2 HIV epitope cluster

Table B.3: The cognate antibodies used to create the HIV epitope cluster. CDRs are determined using the IMGT CDR definition.

PDB code	Chain	CDR1	CDR2	CDR3
3se9	Heavy	EDIFERTEL	VKTVTGAV	ARQKFYTGGQGWYF DL
	Light	SYGH	ATS	QQLEF
4j6r	Heavy	GYSFTDYV	IKPERGAV	ARGVRRDASWWLQF
	Light	QGVGSD	GAS	QQYET
4jdt	Heavy	GYTFTGYY	INPNSGGT	ARGKYCTARDYYNW DFQH
	Light	QSGS	SGS	QQYEF
4jkp	Heavy	GYEFLNCP	LKPRWGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4jpv	Heavy	GYNIRDYF	INPKTGQP	ARQRSDYWDFDV
	Light	NGY	DGS	QVYEF
4jpw	Heavy	GYTFTNYI	IKPVFGAV	ARDESGDDLKWHLH P
	Light	QGIGSS	GAS	AVFQW

PDB code	Chain	CDR1	CDR2	CDR3
4lsp	Heavy	PYWVNPAPEHF	MNPTNGAV	ARAQKRGRSEWAYA H
	Light	RGIGKD	DAS	QQYET
4lst	Heavy	GYEFIDCT	LKPRGGAV	TRGKNCDYNWDFEH
	Light	QYGS	SGS	QQYEF
4lsu	Heavy	GYTFSDYF	MNPQWGQV	ARRMRSQDREWDFQ H
	Light	STS	DGN	NAFEF
4olu	Heavy	GYEFINCP	MKPRGGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4olv	Heavy	GYEFINCP	MKPRFGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4olw	Heavy	GYEFINCP	MKPRHGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4olx	Heavy	GYEFINCP	MKPRLGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4olz	Heavy	GYEFINCP	MKPRWGAV	TRGKYCTARDYYNW DFEH
	Light	QYGS	SGS	QQYEF
4om0	Heavy	GYEFINCP	MKPRYGAV	TRGKYCTARDYYNW DFEH

PDB code	Chain	CDR1	CDR2	CDR3
	Light	QYGS	SGS	QQYEF
4rwy	Heavy	EYTFNEFV	IKRSGRL	ARDGLGEVAPDYRY GIDV
	Light	QGLNF	APS	QEYSSTPYN
4rx4	Heavy	EYTFNEFV	IKRSGRL	ARDGLGEVAPAYLY GIDA
	Light	QGLNF	GPT	QEYSSTPYN
4s1q	Heavy	GYNFREYS	IKGMWGAV	VRKGPSCPHCGDFH WQH
	Light	QYGS	SGS	QQYEF
4s1r	Heavy	GYDFMESL	INPRGGGV	VRGKSCCGGRRYCN GADCFNWDFEH
	Light	QYGS	SGS	QQYEF
4s1s	Heavy	GYTFIKYY	INPRGGQV	VRTADCERDPCK GWVFPH
	Light	QYGS	SGS	QQYEF
4xmp	Heavy	GYEFVEIL	MNPRGGGV	VRGRSCCGGRRHCN GADCFNWDFQH
	Light	QAITPRH	GTS	QCLEA
4xny	Heavy	GYDFMESL	MNPRGGGV	VRGKSCCNGRRYCN GADCFNWDFEY
	Light	QAISKSH	GTY	QCFEG
4xnz	Heavy	GFNFREYS	IKGMWGAV	VRKGPSCPHCGDFH WQH
	Light	QGGNS	DTS	QQFEF

PDB code	Chain	CDR1	CDR2	CDR3
4xvt	Heavy	GYTFLNCP	MKPRGGAV	ARGKYCTASDYYNW DFEH
	Light	QYGS	GGS	QQYEF
4ydi	Heavy	GYTFNAYI	IKPKFGAV	ARDRLYDGSSWRLD P
	Light	QGIGSD	HAS	QVIES
4ydl	Heavy	GYKFIDHF	INPRGGGV	ARGFAGYEWSFI
	Light	QGILSNQ	GGS	QILEF
5cd5	Heavy	GYTFISSF	MNPRHGAV	VTSRTKDYDWDFV
	Light	QRIDNW	KAS	QQFEE
5f96	Heavy	GYTFNDYY	IDPANGRP	VRNVGTAGSLLHYD H
	Light	RSVRNN	DAS	LQYNNWWT
5f9o	Heavy	GYNFNDYY	IDPSGGRT	VRNVGTAGSLLHYD H
	Light	QSVRNN	GAS	QQYNNWWT
5f9w	Heavy	GYTFTNYY	IDPSWGRT	ARNVATEGSLLHYD Y
	Light	QSVRSN	GTS	LQYNNWWT
5fec	Heavy	GYTFTGYY	INPNSGGT	ARERSDFWDFDL
	Light	QDISNY	DAS	QQYEF
5i9q	Heavy	GYTFTDYF	INPLTSQP	ARRHSDYCDFDI
	Light	AGYL	DGS	QVYEF
5igx	Heavy	GYTFTGYY	INPNSGGT	ARGKYCTARDYYNW DFQH

PDB code	Chain	CDR1	CDR2	CDR3
	Light	QSVSSY	DAS	QQYEF
5t33	Heavy	GFSFNEKA	IWYNSIHK	ANAGGGKYDYLDV
	Light	TLSTKW	EGN	FSTDSSGQTWV
5te7	Heavy	GYTFTAHI	IKPQYGAV	ARDRSYGDSSWALD A
	Light	QGVGSD	HTS	QVLQF
5uem	Heavy	GYEFPDYY	IKVGHGGGA	SRDNFGTRPVPGRG YYYGMDV
	Light	YGLDTSH	GTS	QNSGGGTPLI

B.3 Variants used for the SARS-CoV-2 dataset

Antibody name	Non-cognate RBD variant mutations	PDB code of complex	Reference
AB1	E406W N487R K417N, E484K, N501Y	7mjl	Sun et al. (2021)
S2X259	G504D	7m7w	Tortorici et al. (2021)
BG10-19	K417N, E484K, N501Y	7m6e	Scheid et al. (2021)
BG1-22	K417N, E484K, N501Y	7m6f	Scheid et al. (2021)
BG4-25	K417N, E484K, N501Y	7m6d	Scheid et al. (2021)
S2-M28	L18F, R246I, D80A	7ly0	McCallum et al. (2021)
S2-L28	L18F, R246I, D80A, D215G L18F R246G D253G D253Y	7lxx	McCallum et al. (2021)
S2-X333	L18F, R246I, D80A	7lxw	McCallum et al. (2021)

Table B.4: Overview of the SARS-CoV2 variant binding dataset. Note that 7mjl was back-mutated Y501N to restore wild-type, as no complex structure against the wild-type RBD was available. The triple mutant K417N, E484K, N501Y characterises the variant of concern B.1.351.

C | Antibody-antigen binding classification with equivariant graph neural networks

Contents

C.1 Model architectures

C.1.1 GCN

	parameter	value
Model parameters	GCN layers	256, 512, 512, 512
	dropout ratio	0.2
	graph mode	int. nbhd
	node types	atom
	embedding layer	no
	pooling	max
Training parameters	learning rate	0.001
	learning rate scheduler	no
	batch size	32
	max epochs	20
	optimiser	Adam
	loss	binary cross-entropy

Table C.1: GCN model architecture and training parameters

C.1.2 SE(3)-Transformer

	parameter	value
Model parameters	number of GSE3Res blocks	4
	channels in GSE3Res block	32
	attention heads	4
	degrees	2
	dropout ratio	0.2
	graph mode	several (details in methods and results of chapter 6)
	node type	atom
	pooling	max
Training parameters	learning rate	0.001
	batch size	1 (due to high memory requirement)
	max epochs	4
	learning rate scheduler	Cosine annealing with warm restarts
	optimiser	Adam
	loss	binary cross-entropy

Table C.2: SE(3)-Transformer model architecture and training parameters. While the batch size was 1 due to the high memory requirement of the attention weights as well as (depending on the graph generation mode) the input graphs, 32 forward passes of the model were run for each backward pass, making the effective batch size 32.

C.1.3 EGNN

	parameter	value
Model parameters	number of EGNN layers	1, 3, 5 or 7 (details in methods and results of chapter 6)
	channels per EGNN layer	64, 128 or 256 (detail in methods and results of chapter 6)
	embedding	yes, as many channels as EGNN layer
	dropout ratio	0.2
	graph mode	several (details in methods and results of chapter 6)
	node type	atom/residue (details in methods and results of chapter 6)
	pooling	max
	node feature normalisation	Graphnorm (details in methods and results of chapter 6)
Training parameters	learning rate	0.001
	batch size	32
	learning rate scheduler	Cosine annealing
	max epochs	20
	optimiser	Adam
	loss	binary cross-entropy

Table C.3: EGNN model architecture and training parameters

C.1.4 Ten-pose EGNN

	parameter	value
Model parameters	Number of EGNN layers	1, 3 (details in methods and results of chapter 6)
	Channels per EGNN layer	128
	Embedding	yes, as many channels as EGNN layer
	dropout ratio	0.2
	graph mode	interaction edges
	node type	atom/residue (details in methods and results of chapter 6)
	pooling	max
	node feature normalisation	Graphnorm
Training parameters	learning rate	0.0004
	batch size	32
	learning rate scheduler	Cosine annealing
	max epochs	40
	optimiser	Adam
	loss	binary cross-entropy

Table C.4: Ten-pose EGNN model architecture and training parameters