Computational Approaches to Antibody Library Design and Property Prediction

Lewis Chinery

Jesus College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2024

Acknowledgements

I somewhat stumbled my way into this DPhil and am incredibly thankful to have done so. I am especially grateful for the people who have kept me standing along the way.

Firstly, thank you to my family. You have given me every opportunity in life and each bit of encouragement and decision you made has helped shape me into who I am today.

This thesis of course would not have been possible without Charlotte. Your guidance has been instrumental in defining my DPhil and making me a more capable and confident researcher. Thank you also to all my supervisors from GSK - Newton, Iain, TK, and JJ. Your insights have always been useful and it has been brilliant to get a glimpse of the real world of drug discovery.

OPIG has been a fantastic place to spend the last four years. Thank you to everyone I've crossed paths with, especially Alissa, Eve, Lucy, Sarah, and Tom for regularly distracting me from work. I will miss everyone a lot.

Finally, thank you to Priscilla. Your love and support have been unwavering throughout. You invariably make a tough day better and I'm incredibly proud of all you've achieved over the past four years alongside being my full-time project manager and stress guru.

Doing a DPhil is hard at times. Thank you everyone for making it that bit better.

Declaration

I declare that I have performed all the work described in this thesis unless stated otherwise. When others have contributed, these contributors are credited by name.

List of publications

This thesis expands upon the following published paper and pre-prints, completed over the course of my DPhil.

Paragraph - antibody paratope prediction using graph neural networks with minimal feature vectors. **Lewis Chinery**, Newton Wahome, Iain Moal, & Charlotte M Deane. *Bioinformatics*, Volume 39, Issue 1 (2023)

Baselining the Buzz. Trastuzumab-HER2 Affinity, and Beyond! **Lewis Chinery**, Alissa M. Hummer, Brij Bhushan Mehta, Rahmad Akbar, Puneet Rawat, Andrei Slabodkin, Khang Le Quy, Fridtjof Lund-Johansen, Victor Greiff, Jeliazko R. Jeliazkov, & Charlotte M. Deane. *bioRxiv* (2024)

Humatch - fast, gene-specific joint humanisation of antibody heavy and light chains. **Lewis Chinery**, Jeliazko R. Jeliazkov, & Charlotte M. Deane. *mAbs*, Volume 16, Issue 1 (2024)

Abstract

Over the past few decades, antibodies have established themselves as powerful therapeutics, used to treat viral infections, auto-immune diseases, and many cancers. However, designing and optimising such antibody drugs is still costly and time-consuming. In this thesis, we aim to demonstrate how computational methods - both old and new - can improve this process.

First, we introduce Paragraph, our graph-based paratope prediction tool that offers accurate residue-level predictions in just a tenth of a second. Paragraph can help guide computational docking experiments or focus researchers' attention on where optimising mutations should be made. Next, we explore how large language models and other tools can design antibody libraries enriched in high-affinity variants from just a single known binding sequence. We also demonstrate that once a few hundred of these variants have been tested experimentally, we can train simple machine-learning methods on this data to help screen future variants and further enrich our antibody library. Finally, we introduce Humatch, our humanness classifier and humanisation tool. Humatch uses a combination of three highly accurate CNNs and germline data to offer rapid experimental-like humanisation and ensure final heavy-light designs remain well-matched.

All tools and methods presented in this thesis are easily accessible to other researchers. We hope that others may use and build upon these developments to continue to improve antibody therapeutic development.

Contents

1 Introduction

In this chapter, I introduce antibodies and the critical role they play in our natural immune system. I also describe how, over the past four decades, we have engineered antibodies to become powerful drugs, used to treat various diseases including viral infections and cancers. In particular, I focus on how data and machine learning are helping to accelerate the development of these treatments.

Immunology and machine learning are both vast topics, so I will not attempt to cover all aspects of them here. In the following sections, I focus on covering only the main concepts present throughout the rest of this thesis, from amino acids to zero-shot learning.

1.1 Antibodies

1.1.1 Innate immunity

Our immune system is an essential and fantastically complex part of the human body. Every second of every day, our immune system fights to protect us from unwelcome foreign pathogens, such as bacteria and viruses, and our own malfunctioning cells e.g. cancer.

The innate immune system is the first line of defence in these fights. Also described as 'non-specific' immunity, the immune cells involved at this stage mostly recognise a small set of conserved pathogenic targets (Janeway 1989). These targets, or pathogen-associated molecular patterns (PAMPs), include peptides, glycans, and viral double-stranded ribonucleic acid (RNA) (Alberts et al. 2002). Toll-like receptor (TLR) proteins, present on the surface of white blood cells, such as macrophages and neutrophils, recognise and bind these various PAMPs (Kirschning et al. 1998; Yang et al. 1998). This concept of cell-surface proteins binding foreign molecules will be revisited in the next section when antibodies and our adaptive immune system are discussed.

Once bound to their target, our innate immune cells may engulf and destroy the cell, bacteria, or virus displaying the offending molecule in a process referred to as phagocytosis (Tauber 2003). Additionally, recognition of the PAMP may trigger the release of signalling molecules, including cytokines, that help recruit other immune cells to the infected area to continue the fight (O'Shea et al. 2008).

Critically, this response is fast, taking just minutes or hours (Janeway et al. 2001). This speed means that, provided our cells have the appropriate receptors, the innate immune system can eliminate certain diseases before we realise we have them.

Unfortunately, not all disease-causing pathogens or cancerous cells display molecules that can be recognised by our innate immune system. In these instances, we must rely on our adaptive immune system.

1.1.2 Adaptive immunity

While our innate immune system recognises only a small set of conserved disease-related molecules, our adaptive immune system can, as the name suggests, adapt to recognise a theoretically unlimited set of targets if given enough time (Janeway et al. 2001). Additionally, unlike our rapid innate immune response, our adaptive immune response is highly specific to the disease in question, meaning we see few off-target effects. This high specificity is of particular interest to drug developers (see Section 1.2).

There are many different cells involved in adaptive immunity, and they interact with many more from our innate immune system (Gartner et al. 2011). Here, I will focus mainly on the role of B-cells and the antibodies they produce in tackling disease.

1.1.2.1 B-cells

B-cells, like our innate immune cells, are a type of white blood cell that possess receptors on their surface to detect foreign pathogens. In the case of B-cells, these are simply called B-cell receptors (BCRs) and they are the membrane-bound form of antibodies. An individual B-cell will possess hundreds of thousands of BCRs on its surface (Yang et al. 2016) to maximise the chances of being able to bind its target antigen (*anti*body-*gen*erating molecules, synonymous to PAMPs). All the BCRs on its surface will be identical to one another, however, different B-cells will often have unique BCRs from one another (Hoehn et al. 2016).

As humans possess a circulating repertoire of approximately 10^{12} B-cells (Alberts et al. 2002; Yaari et al. 2015) - more than the number of stars in the Milky Way galaxy - and many of these will display unique BCRs, we are almost certain to have at least one that will bind to an antigen with low affinity (interaction strength) for any prospective pathogen. This wide net cast by our adaptive immunity is critical for our protection and originates from a process known as V(D)J-recombination.

1.1.2.2 V(D)J-recombination and B-cell diversity

The instructions for building most proteins in our body exist in single places in our DNA (Crick et al. 1961). However, the instructions for building each protein chain that together form our BCRs are spread throughout our genome in various gene segments and across different chromosomes (Hozumi et al. 1976). Most relevant to this thesis are the Variable (V), Diversity (D), and Joining (J) immunoglobulin genes (see Figure 1.1.1). These genes encode different structural sections of each BCR chain and many variations of each gene exist spread across our chromosomes (Honjo 1983; Tonegawa 1983).

To build a functioning BCR, our cells first need to recombine one segment of each of the V, J, and (for some chains) D gene segments into a single continuous stretch of DNA. This breaking and rejoining of chromosomes to form continuous V(D)J stretches happens as new B-cells form. The combinatorial V-(D)-J diversity in this process is high given the ∼150 functional gene segments in our genome - the exact number varies for different people (Janeway et al. 2001). This diversity is further amplified through junctional diversity, where small numbers of nucleotides (the building blocks of DNA) are added or removed where V, D, and J genes are joined (Roth 2014). The outcome of this combinatorial and junctional diversity is a nearly limitless variation of 'germline' BCRs capable of recognising all antigens.

The human body however is not limitless - as mentioned in Section 1.1.2.1, our bodies contain approximately 10^{12} B-cells which is not enough to guarantee one will bind every disease target *strongly*. High-affinity binding must therefore be achieved through another process of our adaptive immune system - somatic hypermutation.

1.1.2.3 Somatic Hypermutation and affinity maturation

Somatic hypermutation (SHM) is a process that only occurs in B-cells once they have bound an antigen (Bernard et al. 1978; Griffiths et al. 1984). Once activated by antigen binding, these B-cells rapidly divide, around once every six to eight hours (Janeway et al. 2001). During each cell division, random mutations are introduced to certain areas in the BCR encoding region (Complementarity Determining Regions, CDRs, see Figure 1.1.1) of the genome at a rate approximately 1,000,000 times higher than normal (Levy et al. 1989; Milholland et al.

2017). This propensity for 'errors' to occur during cell division translates into small changes (typically one or two amino acids) to the BCRs of the next generation of the original B-cell.

Clonal selection (positive feedback) then occurs, where all variants of the precursor compete to bind the antigen and those that bind strongest proliferate and divide more often (Burnet 1957). This process repeats itself, with B-cells introducing more and more random mutations until high-affinity binding is achieved, typically taking a couple of weeks.

1.1.2.4 B-cell differentiation and antibody production

Once a naive B-cell has undergone affinity maturation, the matured clones differentiate into various B-cell types (MacLennan 1994), mainly plasma cells or memory B-cells (MBCs).

Like naive B-cells, MBCs possess BCRs on their surface. However, they divide rarely (if ever) and live far longer than naive B-cells, often, many years (Crotty et al. 2003; Yu et al. 2008). MBCs are the source of our long-lived immunity and typically result in us experiencing less severe disease when exposed to repeat infections by the same pathogen (Jenner 1798; Ahmed et al. 1996). The lower severity is due to our adaptive immune system being able to bypass V(D)J-recombination and SHM, offering a faster immune response than previously possible.

Plasma cells meanwhile look and behave very differently from naive B-cells. Instead of possessing many surface BCRs, plasma cells secrete antibodies, up to 2,000 every second (Alberts et al. 2002). These antibodies look identical to and have the same binding properties as the affinity-matured BCRs. Secreting antibodies has the significant benefit of enabling the recognition and binding of many more pathogen copies than individual B-cells are capable of alone.

Once an antibody binds to an antigen, the pathogen it belongs to can suffer from reduced mobility, increased 'clumping' (agglutination), and a weakened ability to bind and harm healthy cells in our body. Furthermore, once bound by an antibody, the pathogen becomes 'visible' to our innate immune system thanks to the conserved antibody constant region (F_c) that does not bind and faces away from the pathogen (see Figure 1.1.1). This new recognition via the bound antibody (opsonisation) allows innate immune cells, such as macrophages, to engulf and destroy the pathogen as described in Section 1.1.1 (Alberts et al. 2002).

If the adaptive immune process works as intended then the disease cause will be eliminated, the specific high-affinity antibodies will degrade, typically in around three weeks (Ko et al. 2021), and we will possess some lasting immunity against the disease should we face it again via MBCs.

1.1.3 Antibody structure

The previous sections introduced antibodies and their role in tackling diseases. So far, antibodies have been described as multi-chain proteins with one part that binds strongly and specifically to an antigen, and another that is constant and offers our innate immune cells something to recognise and bind. This section describes antibody structure in more detail.

1.1.3.1 Primary structure (sequence)

Antibodies are composed of four protein chains - two identical 'heavy' chains and two identical 'light' chains, that are shorter in length (see Figure 1.1.1). These protein chains are each composed of sequences of amino acids, often referred to as 'residues'. In nature, we observe 20 unique 'canonical' amino acids, each assigned a single-letter code e.g. 'A', 'C', 'D', etc. This amino acid alphabet allows protein chains to be written as a string of letters, known as their primary structure e.g. `VYIHPF` describes the six residues that make up the small protein Angiotensin IV, a hormone that increases blood pressure (Wright et al. 1995).

Antibody heavy and light chains are much longer than typical protein hormones. Each heavy chain contains approximately 450 residues, while light chains contain approximately 220 (Janeway et al. 2001). Tens or hundreds of these residues can vary between two antibodies produced by different plasma cells, though most antibody regions are highly similar. This consistency is due to their encoding by a small number of Constant (C) genes which remain unaffected by V(D)J-recombination and SHM. Nevertheless, thanks to the library of 20 canonical amino acids, even small numbers of mutation sites can result in huge antibody variety - for a protein with n mutation sites, there are 20^n possible unique sequences that could be formed. Just 64 mutation sites allow more combinations than the number of atoms in the observable universe.

The amino acids that vary between antibodies tend to be concentrated in three well-defined

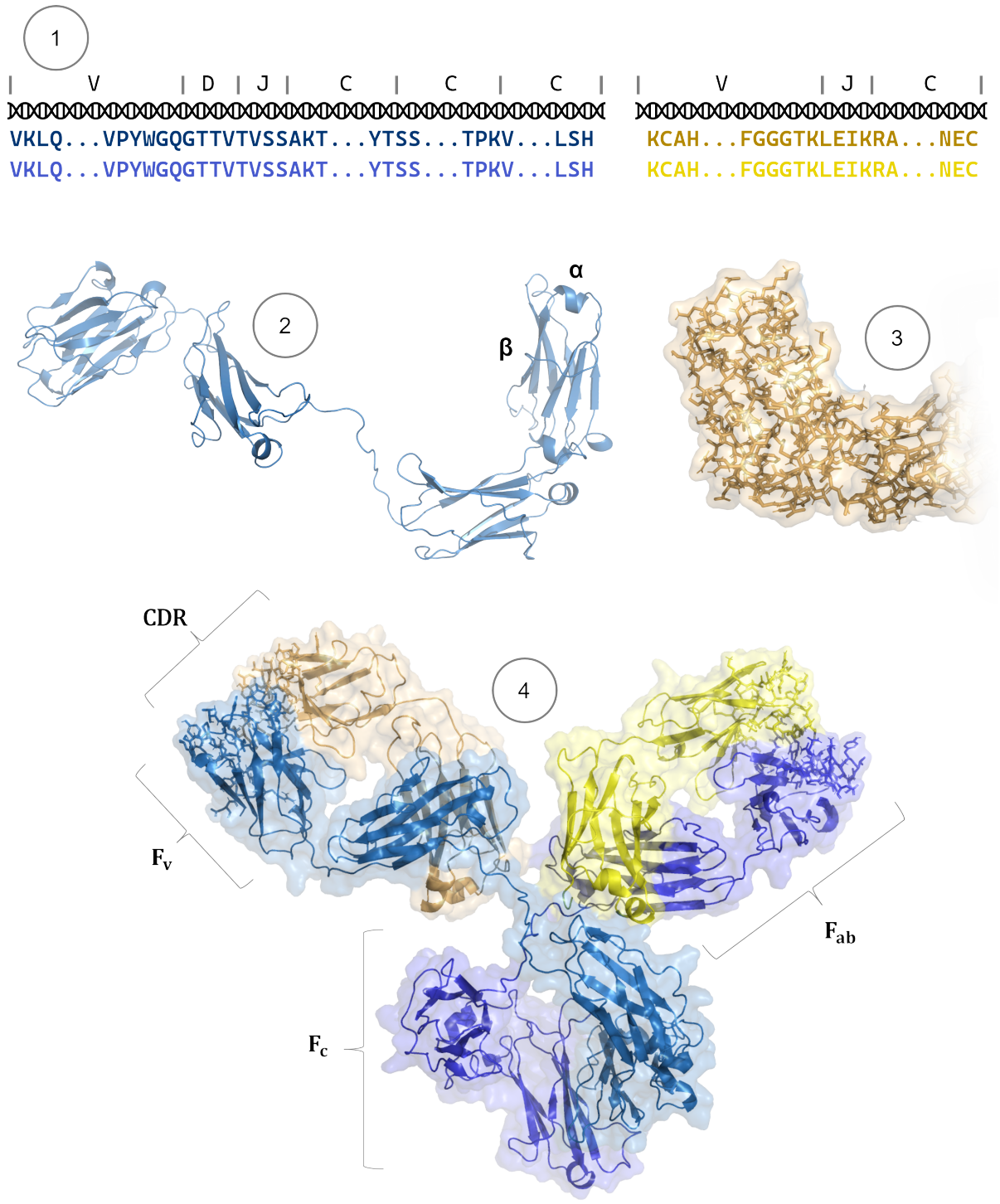

Figure 1.1.1: An overview of an antibody's structure. **1)** An antibody is formed of four protein chains - two identical heavy chains (blues) and two identical light chains (yellows). These chains are encoded by different gene segments within our DNA - Variable (V), Diversity (D), Joining (J), and Constant (C) regions. The sequence of amino acids (shown as their single-letter codes) forms each chain's primary structure. **2)** A 'cartoon' representation of a heavy chain's secondary (backbone) structure, including beta sheets (β, flat arrows) and alpha helices (α, corkscrew arrows). **3)** A 'sticks' and translucent surface representation of a light chain's tertiary structure that includes all backbone and side-chain atoms of the constituent amino acids. **4)** The quaternary (complete) assembled Y-shaped antibody structure (side chains mostly omitted for clarity). The fragment-variable region (F_v) is formed by the heavy and light chains' V, (D), and J-gene encoded regions. Six CDR loops (three per chain shown as sticks) are found on the end of each F_v. These loops dominate antigen binding. The fragment-antigen-binding (F_{ab}) region includes the F_v and the amino acids encoded by the first constant region of each chain. The fragment-constant (F_c) is formed of the final two heavy chain constant regions.

regions on each chain called the Complementarity Determining Regions (CDRs). The CDRs vary most as nucleotide insertions and somatic hypermutations are focused here (see Sections 1.1.2.2 & 1.1.2.3). Once each protein chain folds into a three-dimensional (3D) structure, these regions form loops that dominate antigen binding (see Figure 1.1.1).

1.1.3.2 Secondary, tertiary, and quaternary structure (3D)

Antibodies do not exist as unstructured 'loose' chains. Instead, each chain consistently folds into a well-defined 3D structure. Once folded, both sets of heavy and light chains bind, resulting in an antibody's characteristic Y-shape (see Figure 1.1.1).

All proteins' 3D structures, including antibodies, can be described at different levels (Linderstrom-Lang 1952). The secondary structure describes the protein 'backbone'. This backbone consists of the three atoms common to all amino acids - two carbons, and one nitrogen. Together, the backbone atoms trace a single line following the twists and turns of a protein chain. These backbone atoms may form unstructured loops but often form common, consistent secondary elements across large portions of the protein, mainly alpha helices and beta sheets (see Figure 1.1.1).

The tertiary structure of a protein describes the full 3D structure of a single protein chain, including the backbone and 'side chain' atoms. For multi-chain proteins, like antibodies, the quaternary structure is the final complex of all tertiary structures (see Figure 1.1.1). Knowledge of the final 3D structure of an antibody is often of great interest to researchers given proteins' strong link between their structure and function (Hegyi et al. 1999).

Antibodies' main function is binding their antigen with high affinity and specificity. This binding is almost always primarily determined via the six CDR loops found at the end of each fragment variable region (F_v), composed of the variable heavy (VH) and variable light (VL) sequences (MacCallum et al. 1996). As this region dominates binding, most antibody research is focused on better understanding the structure, function, and properties of the F_v and CDR loops. To make this research easier, it is useful to first have a common scheme to identify the CDR loops of different antibodies that may have highly variable composition and lengths.

1.1.3.3 Antibody numbering

Earlier, it was mentioned that antibody heavy and light chains contain approximately 450 and 220 residues respectively. However, the total length of each chain can vary by tens of amino acids largely due to great variation in their CDR loops. Antibodies' differing lengths mean that it is not possible to make a simple one-to-one sequence comparison of different heavy and light chains without first aligning them such that the start, end, and conserved regions match well. This is true even for antibodies of the same length as length differences across the three CDR loops may cancel each other out.

To help with this comparison, antibody 'numbering schemes' have been introduced that offer consistent numbering to highly conserved residues and make it simple to identify the highly variable CDR loops. Several numbering schemes exist and are used by different researchers, including 'Kabat' (Kabat 1983), 'Chothia' (Chothia et al. 1987), 'AHo' (Honegger et al. 2001), 'IMGT' (Lefranc et al. 2003), and 'Martin' (Abhinandan et al. 2008). These numbering schemes offer broad agreement in their CDR definitions and their placement of residue insertions and deletions but differences do exist (Dondelinger et al. 2018). For simplicity, this thesis uses IMGT numbering definitions throughout.

IMGT numbering was first introduced over 20 years ago and applies the same numbering logic to both heavy and light chains. The numbering of each chain within the F_v region runs from '1' to '127' or '128' depending on whether the chain is light or heavy (see Figure 1.1.2). Residues are generally numbered incrementally between these start and end values (e.g. 1, 2, 3, ...) with insertions and deletions focused in the three CDR loops. The first CDR loop (CDR1) is defined to start and end at IMGT positions 27 and 38, the CDR2 loop at positions 56 and 65, and the longest and most variable CDR3 loop at positions 105 and 117. If an antibody has a short CDR loop, certain position numbers may be missing in the middle of the loop (e.g. ..., 56, 57, 65, ...). For longer loops, particularly the CDR3, insertions are defined to exist with letter suffixes (e.g. ..., 111, 111A, 112B, 112A, 112, ...). Convention states that all IMGT insertions should fall between positions 111 and 112 but since its conception examples have been discovered where insertions must appear at different positions in the antibody 'framework' (F_v residues that do not belong to the CDR loops).

Figure 1.1.2: Example antibody variable heavy (VH) sequence numbered and aligned using the IMGT numbering scheme. Framework residues are shown in black and residues belonging to the three CDR loops are highlighted in red. Within the third CDR loop, more insertion positions are possible between residues 111 and 112 but these have been omitted for clarity as they are not present in the example sequence.

Different computational tools exist that allow users to input an antibody sequence of amino acids and output the corresponding numbering. ANARCI (Dunbar et al. 2016a) is a popular open source numbering tool that aligns input sequences to a set of annotated IMGT germline sequences. This alignment uses Hidden Markov Models to score how likely sequential amino acids are to appear at each position for every V- and J-gene (accounting for insertions and deletions). The most significant alignment is then used to number the input sequence. ANARCI has provided all IMGT numbering used throughout this thesis.

1.1.4 Antibody-antigen binding

So far, antibodies have been described largely in isolation. However, researchers are often interested in the process of antibody-antigen binding and the final bound 3D complex.

1.1.4.1 Binding affinity

One aspect of antigen-antibody binding that is of interest is binding affinity i.e. how strongly an antibody (Ab) and antigen (Ag) stick together. This affinity is typically measured using a disassociation constant, K_D.

Equation 1.1 describes the two-way process of antibody and antigen binding (left to right) and dissociation (i.e. unbinding, right to left). Given enough time, a mixture of antibodies and antigens in fixed conditions (e.g. pH, temperature etc.) will reach an equilibrium, when the rate of the left and right processes are the same. The equilibrium concentrations of each of the

separate and joined states when this is achieved are given as $[Ab]$, $[Ag]$, and $[AbAg]$ in Equation 1.2.

$$Ab + Ag \rightleftharpoons AbAg \tag{1.1}$$

$$K_D = \frac{[Ab][Ag]}{[AbAg]} \tag{1.2}$$

For antibodies and antigens that bind with high affinity, we expect most to be bound together in equilibrium i.e. the denominator of Equation 1.2 would be large. K_D is therefore inversely related to affinity. As $[Ab]$, $[Ag]$, and $[AbAg]$ are typically measured as molar concentrations (M, moles per litre), K_D also shares the same units (one mole of Ab or Ag equals 6.022×10^{23} copies of either molecule). Typical K_D values of antibodies raised in an adaptive immune response vary between 10^{-7}M (low affinity) and 10^{-11}M (high affinity) (Khani-Habibabadi et al. 2024). In drug discovery campaigns, single-digit or below nano-molar (nM) K_D values are typically targeted (Roskos et al. 2004).

Measuring antibody-antigen binding affinity is now widely possible, though the availability of open source public data is still limited. High-throughput methods, such as Fluorescence-Activated Cell Sorting (FACS, described in more detail in Chapter 3 and the Appendix) allow the binary separation of hundreds of thousands of binding and non-binding antibodies for a given antigen (Doerner et al. 2014). For a higher cost, Surface Plasmon Resonance (SPR) and Biolayer Interferometry (BLI) experiments can provide exact K_D measurements for hundreds of antibodies (Kamat et al. 2017). However, none of the above methods offer insight into the orientation of the bound antibody-antigen complex.

1.1.4.2 Paratope and epitope

By design, engineered and natural, mature antibodies are highly specific, often binding to only a single site on one antigen with high affinity. The antibody residues that bind the antigen are called the paratope and are dominated by CDR residues (MacCallum et al. 1996). The corresponding part of the antigen is referred to as the epitope (see Figure 1.1.3). Paratope and epitope residues are typically separated by less than 5 Angstroms (Å, 5×10^{-10}m), corre-

sponding to the approximate maximum attractive distances of the hydrogen bonds and polar, hydrophobic, and Van der Waals interactions that dominate binding.

Different antibodies may bind to different or similar epitopes on a given antigen (e.g. a viral protein). Antibody mutations that occur in the paratope are much more likely to affect binding than those that occur outside (though these could affect the global structure of the antibody and still influence binding) (Shannon et al. 1999). Similarly, mutations to the epitope can disrupt antibody binding and thus we see these positively selected by viruses to escape detection by our immune systems (Keck et al. 2018; Starr et al. 2021). When designing antibody therapeutics, researchers may therefore prefer an antibody that targets a well-conserved epitope over one with higher affinity that targets an epitope more susceptible to mutations.

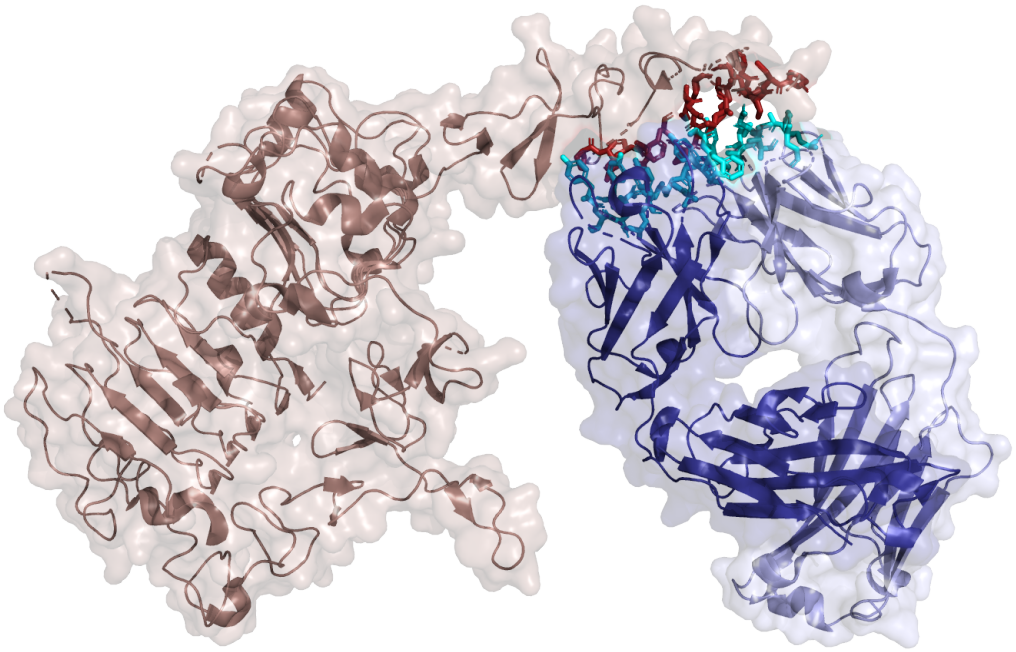

Figure 1.1.3: An antibody F_{ab} shown bound to its antigen. The heavy chain is shown in dark blue, the light chain in light blue, and the antigen in brown. The paratope and epitope residues have been highlighted as 'sticks' representations in cyan and red, respectively.

1.2 Antibodies as therapeutics

Antibodies play a crucial role in our adaptive immune system, offering specific, high-affinity binding of disease targets. These properties and biological modes of action (e.g. opsonisation, agglutination, etc., see Section 1.1.2.4) are also highly sought after in drug candidates to provide patients with potent treatments that yield few side effects.

1.2.1 The market for antibody therapeutics

The first antibody drug, approved in 1986, was developed to reduce kidney transplant rejection rates (Mullard 2021). The 100^{th} monoclonal antibody (mAb, identical antibodies produced from the same cell lineage) therapeutic was approved by the United States Food and Drug Administration (FDA) in 2021 (Mullard 2021). Since then, the number of approvals has continued to grow.

A big attraction of using antibodies as therapeutics is their application to different diseases. Last year, twelve new mAb therapeutics were approved by the FDA for the treatment of viral infections, autoimmune diseases, Alzheimer's, and various cancers (Torre et al. 2024).

Despite recent increases in sales of non-antibody treatments, such as COVID-19 vaccinations and diabetes/weight-loss drugs, mAbs continue to rank highly in the list of best-selling therapeutics. In 2023, Keytruda (pembrolizumab) and Humira (adalimumab) topped this list with combined annual sales of almost $40bn (Buntz 2024).

1.2.1.1 Engineered antibody formats

Since the approval of the first antibody therapeutic, researchers have sought to engineer novel antibody formats to increase their effectiveness and range of applications (Jin et al. 2022). These developments include bispecific antibodies, where each antibody arm is engineered to bind a different epitope. Bispecifics are of particular use in cancer therapy where one arm binds the cancer cell and the second recruits other immune cells more specifically than a standard F_c (Kantarjian et al. 2017; Trabolsi et al. 2019).

Single-chain variable fragments (scFvs) are single-chain versions of an antibody F_v where the VH and VL are joined by a short protein chain linker. scFvs have the advantage of being easier and cheaper to produce than full-length antibodies given their simpler structure (Ahmad et al. 2012). Unfortunately, they can also be less stable than full-length antibodies and present greater immunogenic risks as they do not occur naturally in humans (Wen et al. 2013).

Antibody-drug conjugates (ADCs) are antibodies linked to other drug molecules. ADCs facilitate the targeted delivery of the additional drug molecule to a particular site, such as a cancer cell, helping to reduce potential side effects (Sievers et al. 2013).

Many more antibody formats exist and continue to be developed, and research is even underway to improve their ability to penetrate cells and target intracellular antigens (Gaston et al. 2019). This thesis focuses mainly on traditional mAb therapeutics. However, the methods and aims presented later, such as affinity optimisation and humanisation, remain important regardless of the antibody format.

1.2.2 The drug discovery pipeline

Developing new antibody drugs typically takes over a decade and requires billions of dollars of investment (Hughes et al. 2011). This process can be broadly split into 'discovery' and 'development' stages (Ain et al. 2020). The first of these includes pinpointing promising drug targets, identifying initial drug candidates, and optimising these candidates to meet desired criteria (see Figure 1.2.1). The latter stage involves pre-clinical and clinical studies, where the optimised drug is administered to patients in different doses and compositions to verify its efficacy and safety in real-world settings.

Both stages of this pipeline require considerable time and investment and the attrition rate of promising leads at each step is high. This thesis focuses on offering time and cost savings to the initial discovery stage of the pipeline only. Computational prediction and optimisation of different antibody properties allow more candidates to be screened *in silico* (computationally) than would ever be possible experimentally. This screening aims to ensure only the best leads make it into the lab, reducing the attrition rate and increasing the probability of higher

downstream success rates in clinical trials.

1.2.3 Lead identification

The methods presented in this thesis rely on first having an initial antibody lead - an antibody that binds the target of interest, but perhaps with low affinity and high immunogenicity. Currently, computational tools cannot reliably design these leads from scratch (Hummer et al. 2022), so we rely on experimental techniques to identify them.

One way in which initial antibody leads might be identified is via high throughput library screens of individuals who have raised an immune response to the disease of interest e.g. SARS-CoV-2 (Williams et al. 2017; Xiao et al. 2019). In these cases, serum from a patient is extracted and over-expressed antibody clones are picked out as likely candidates. Copies of these antibodies can then be made in the lab and validated as binding or not binding to the target antigen.

In many cases though, the above is not possible - cancers are deadly as our bodies do not mount an immune response against these cells. In these instances, initial leads might be identified by selecting an antigen you wish to target and administering this to a mouse, or other animal (Gupta 2017; Schardt et al. 2024). High-affinity antibodies raised by these animals can then be extracted similarly to above, though without further humanisation these antibodies would lead to immunogenic reactions in humans.

Finally, *in vitro* methods (those that take place outside living organisms i.e. 'test tube' methods), such as yeast and phage display can be used to identify leads (Winter et al. 1991; Hoogenboom 2005). With these display methods, many yeast or bacteriophage cells are engineered to express a variety of human antibodies. Target antigens are then screened against these synthetic antibody libraries to identify binders. Experimental *in vitro* display techniques continue to improve and can screen fully human antibodies that require no humanisation. However, these libraries are not currently as vast or adaptable as natural immune responses (Almagro et al. 2019), so the chances of identifying a high-affinity lead are lower than other methods.

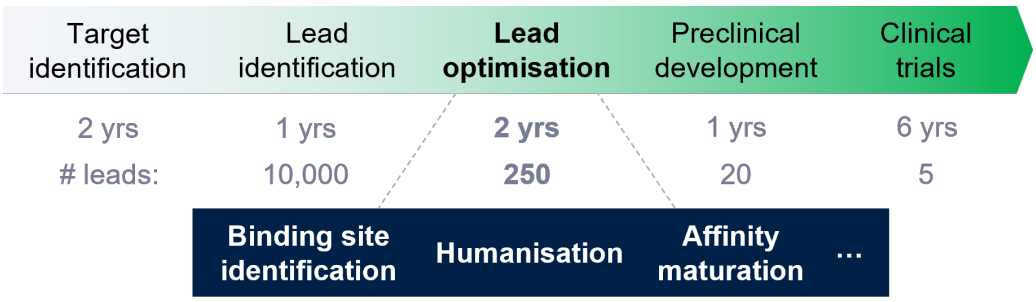

Figure 1.2.1: An overview of the drug discovery pipeline (green, left to right) including the approximate time required to complete each step. The number of leads that typically progress from each stage are also shown. This thesis explores methods to improve antibody lead optimisation. A non-exhaustive list of lead optimisation goals that we address in this thesis is shown in the blue box.

1.2.4 Lead optimisation

Once an initial lead is identified, much time and effort is spent optimising it (Wang et al. 2021). This is a tricky, multi-objective problem, as the final antibody drug must have high affinity to the target antigen, avoid binding anything else (off-target effects), have low immunogenicity (side effects caused by our immune system attacking the drug itself), and have good developability parameters such as high stability and low aggregation (Tiller et al. 2015). This thesis does not address all of these areas. Instead, we focus on better understanding the antibody binding site and improving the affinity and humanness of antibodies.

1.2.4.1 Binding site identification

Knowledge of the paratope is of interest to researchers as it allows more constrained, accurate antibody-antigen docking simulations (Ambrosetti et al. 2020b). This information can also inform where optimising mutations are made - mutations targeting improved affinity should be focused in the paratope, while others targeting higher humanness or developability should avoid it (Shannon et al. 1999; Ramon et al. 2024).

Currently, our most accurate knowledge of antibody and antigen binding sites is from crystal structures (Dunbar et al. 2014; Schneider et al. 2022). Crystal structures are examined using X-ray diffraction experiments, where X-rays are fired at a crystal composed of bound antibody-antigen complexes. Electron densities are calculated from the intensities and angles of the diffracted X-rays, offering atomic-level resolution. Crystallising antibody-antigen complexes is challenging though, sometimes taking years and requiring considerable investment due to

experimental trial and error (Boulot et al. 1988; Malia et al. 2011).

Other techniques, such as cryo-electron microscopy (cryo-EM) and Nuclear Magnetic Resonance (NMR), can also provide binding site information without requiring antibody-antigen crystallisation. Cryo-EM involves rapidly freezing a sample of antibodies, firing electrons at the sample, and then measuring and interpreting the scattering pattern (Wang et al. 2017). NMR experiments on the other hand place an antibody sample in a strong magnetic field, aligning the magnetic 'spins' of the atomic nuclei. A resonator then fires radio waves at the sample, 'flipping' the spins, and the subsequent energy released when they 'relax' back to their starting alignment is measured (Hu et al. 2021). Cryo-EM is becoming increasingly popular for studying antibodies as it images antibodies closer to their 'native' state (Wang et al. 2017). However, both cryo-EM and NMR have historically provided lower-resolution structures than X-ray experiments (Dunbar et al. 2014; Schneider et al. 2022), so we use only the latter in this thesis for labelling paratope residues.

1.2.4.2 Affinity optimisation

Once the paratope is known, or approximated by an antibody's CDR loops, this information can be used to improve its affinity to the target antigen.

Deep Mutational Scanning (DMS) is one experimental technique used to improve affinity (Whitehead et al. 2012). In DMS experiments the aim is to carry out all possible single-point mutations within the paratope or area of interest. These single-point variants are then tested for binding against the target antigen and compared against the original sequence. Mutations that maintain or improve binding are preferentially selected in designing multi-point variants.

DMS succeeds in designing high-affinity variants more efficiently than random trial and error (Mason et al. 2021; Shanehsazzadeh et al. 2023). However, antibody-antigen affinity landscapes are not smooth, so combining two beneficial single-point mutations is not guaranteed to produce a strong binder, and vice versa (Hie et al. 2023). Computational methods can aid in this process by helping to screen out low-affinity designs before they are tested (Hie et al. 2023; Shanehsazzadeh et al. 2023). This screening ensures experimental resources are only spent val-

idating the most promising leads and allows the exploration of a greater proportion of sequence space.

1.2.4.3 Humanisation and immunogenicity

Most therapeutics are not genetically human in origin. Instead, many are murine (originating from mice) or obtained from chimeric animal models (Gordon et al. 2024).

Deriving precursor therapeutic antibodies from non-human natural immune responses can ensure high antigen affinity with CDR loops optimised through V(D)J recombination and SHM. However, the framework and constant regions of the antibody that do not bind to the antigen will almost certainly need to be altered before they are safe to give to humans (Hwang et al. 2005). This humanisation process is critical as our immune system will recognise these areas as 'non-self' and attack them (Janeway et al. 2001). Such an attack will lessen the effectiveness of the therapeutic and lead to inflammation and possibly other side effects (Hwang et al. 2005).

Traditional humanisation processes involve grafting the optimised CDR loops onto a human antibody framework (Jones et al. 1986; Riechmann et al. 1988). Several framework residues key to maintaining the CDR structure are then back-mutated to maintain affinity. Alternatively, iterative mutations of the original antibody can simply be made towards the human germline (Pedersen et al. 1994; Roguska et al. 1994).

Both traditional approaches involve experimental trial and error and can still lead to high anti-drug antibody levels (ADAs, immune antibodies raised against the therapeutic antibody) (Marks et al. 2021). In some cases, excessive numbers of mutations may also be made to ensure lower immunogenicity risks at the expense of affinity or developability criteria. The goal of computational humanisation tools then is to suggest an optimised, likely smaller, set of mutations that should yield antibodies with comparable or lower immunogenicity risks.

1.3 Computationally-aided antibody design and optimisation

In the previous two sections, antibodies were described as both natural immune proteins and popular therapeutics. Experimental optimisation techniques were described and shortcomings, mainly high costs and development times, were highlighted. Fortunately, computational methods can offer some improvements to these critical steps of the drug discovery pipeline.

1.3.1 Classical computational design tools

Though deep learning methods, such as AlphaFold (Jumper et al. 2021), have grabbed recent headlines, computational tools have been an essential part of the drug discovery pipeline for decades. Related to this thesis are observational frequency methods such as BLOSUM matrices (Henikoff et al. 1992), sequence clustering approaches such as clonotyping (Li et al. 2001; Yaari et al. 2015; López-Santibáñez-Jácome et al. 2019), and physics-based tools such as docking (Pierce et al. 2014; Zundert et al. 2016).

Observational frequency methods rely on obtaining accurate, representative data for the problem of interest. For example, BLOSUM matrices (BLOcks SUbstitution Matrices), first introduced by Steven and Jorja Henikoff in 1992, measure how often each amino acid is found to be substituted by every other amino acid in conserved, aligned regions of natural proteins. Fortunately, protein sequence data is plentiful (Bateman et al. 2015; Kovaltsuk et al. 2018; Olsen et al. 2022b), so BLOSUM matrices offer a reliable indication of which amino acids are physically and chemically similar. This knowledge allows drug designers to offer sensible mutations to a precursor therapeutic that deliver greater success rates than random mutations alone.

Clustering methods are also widely used to predict which proteins will behave most similarly based on how closely their sequences of amino acids match. Though sequence-similar proteins are not guaranteed to have similar structures and functions, such as high binding affinity, proteins with high sequence overlap (e.g. 90%+) are most likely to. Clonotyping extends simple percentage sequence clustering methods by also requiring each antibody to have similar

V, D, and/or J gene origins. Nowadays, machine learning methods' training and test data are often split by clonotype to reduce 'data leakage' (testing on data very similar to the training data) and to ensure a robust final model (Kapoor et al. 2023; Bernett et al. 2024).

Finally, physics-based methods that aim to simulate real physical forces *in silico* still offer more accurate predictions of binding sites and drug properties than many recent machine learning methods (Buttenschoen et al. 2024). Molecular Dynamics (MD) simulations provide atomic-resolution predictions of antibody properties though at a high computational cost (Case et al. 2005; Phillips et al. 2005; Brooks et al. 2009; Abraham et al. 2015). This high computational cost arises from simulating all physical forces on each atom every femtosecond (fs, 10^{-15}s). The iterative updating of atomic positions and forces means sub-microsecond (μs, 10^{-6}s) simulations can take many weeks to run.

Physics-based docking methods provide faster but less granular predictions of bound antibody-antigen complexes than MD by exhaustively screening thousands of possible poses (Pierce et al. 2014; Zundert et al. 2016). Each pose is then scored based on its predicted minimum energy, with lower energies indicating more stable conformations. Both MD and classical docking methods are widely used in academia and industry, though faster, less accurate, machine-learning approximations of these methods are also being introduced (Ketata et al. 2023; Williams et al. 2023). Moreover, classical tools may be implemented in hybrid with ML methods by, for example, first predicting the antibody and antigen binding sites before docking to reduce the number of pose searches required during exhaustive screening (Ambrosetti et al. 2020b).

In the rest of this thesis, I will use some of these classical computational approaches to baseline machine learning methods. Offering robust, sensible baselines is critical to ensure ML methods learn deeper information from the data they are trained on and validate which metrics are most useful for evaluating different tasks.

1.3.2 Data available for training machine learning models

Classical computational design tools, such as BLOSUM matrices, rely on the availability of plentiful data that accurately describes the problem of interest. Machine learning methods rely

on this data too and as these ML models continue to grow so does the demand for more training data.

1.3.2.1 Sequence data

Earlier, antibodies were described by their sequence of amino acids (primary structure) and their 3D atomic coordinates (quaternary structure, see Section 1.1.3.2). Obtaining protein sequences is much faster and cheaper than their 3D structures. Billions of protein sequences have been recorded and processed in various databases. In this thesis, most tools have been trained on data from the Observed Antibody Space (OAS) (Kovaltsuk et al. 2018; Olsen et al. 2022b) database. Some general protein tools we compare against though, such as Meta's ESM (Rives et al. 2021), explained in detail later, have been trained on wider protein databases such as UniProt (Bateman et al. 2015).

The OAS database contains over two billion antibody sequences (unpaired VH/VL), collated from ∼100 different experiments. The amount of sequence data available to train machine learning models is therefore large, but, as described earlier (see Sections 1.1.2.1 & 1.1.3.1) so is the theoretical total antibody sequence space. Additionally, some sequences in OAS are redundant (identical to other sequences in the database) or incomplete (missing the start and/or end of the VH or VL sequence).

Moreover, most of the data in OAS is unpaired - as of September 2024, OAS only contains two million paired antibody sequences (∼75% originates from just one study), three orders of magnitude less than the unpaired data. This disparity exists because obtaining paired VH/VL data requires more expensive experimental equipment that separates single B-cells and tags their RNA with a unique 'barcode' identifier (Tan et al. 2014; Stoeckius et al. 2017). Nevertheless, unpaired and paired data from OAS and other sources have proved useful in training many machine learning models to high accuracy, including Large Language Models (LLMs), similar to ChatGPT.

To train ML models on antibody sequence data, the data must often first be consistently aligned according to an antibody numbering scheme (see Section 1.1.3.3). Once aligned, each

residue and 'pad' token can be represented with a numerical 'feature vector' (see Figure 1.3.1). This feature vector could be 'one-hot' (a 20D vector consisting of 19 'zeros' and a single 'one' whose placement varies depending on the amino acid in question) or a more complex, descriptive physio-chemical feature vector. These sequence encodings form the basis for many ML methods, but for some problems structure-based ML methods may be more appropriate (see Figure 1.3.1).

1.3.2.2 Structural data

Obtaining the crystal structure of an antibody is far more expensive and time-consuming than elucidating its sequence (Kovaltsuk et al. 2018). As of September 2024, the Structural Antibody Database (SAbDab) (Dunbar et al. 2014; Schneider et al. 2022) contains ~9,000 antibody complexes, five orders of magnitude lower than OAS. Like OAS, SAbDab contains redundancy as certain antibodies and targets have been studied more than others. Some structures are

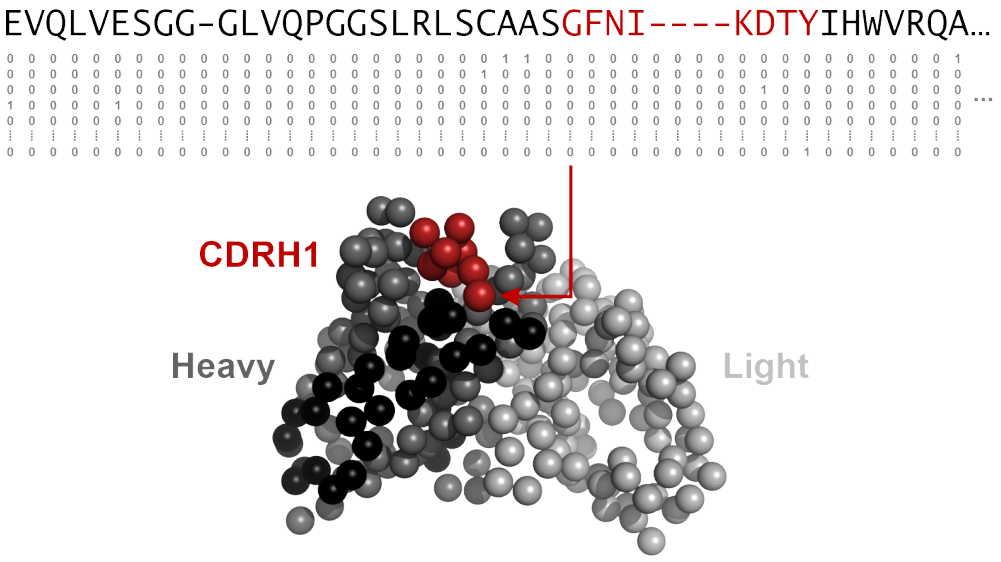

Figure 1.3.1: An overview of common sequence and structure antibody representations. Antibody **sequences** can be numbered and padded using an antibody numbering scheme (top). The heavy sequence shown is truncated for plotting clarity and the CDRH1 residues are highlighted in red. Each amino acid may be represented with a one-hot feature vector (grey), though other features may be used as desired. Once encoded, sequence-based ML networks can be trained on this data in a supervised fashion, if labels are available, or in an unsupervised fashion by masking residues and predicting their identity. **Structure** representations of antibodies are also possible (bottom). Only the antibody F_v is shown. Black and red residues match the sequence above; other heavy and light residues are shown in shades of grey. In this example, the antibody 3D structure is shown as a graph, with one node representing each residue. Edges can be defined to exist between nodes within a certain distance of one another (e.g. 10Å) though these have been omitted for plotting clarity. Once a structure representation has been decided, ML networks can also be trained on these inputs.

also low resolution (e.g. >3Å). Low-resolution structures are less useful for machine learning applications due to the lack of an accurate 'ground truth'. All this means that our knowledge of antibody structures, especially their highly variable CDR loops, is incomplete.

The Protein Data Bank (PDB), which collates all publicly available protein structure data, contains over 200,000 structures (Burley et al. 2019). This resource has been used to train general protein models, such as ProteinMPNN (Dauparas et al. 2022) (see Chapter 3). The PDB also provides data used to pre-train antibody-specific tools, such as PECAN (Pittala et al. 2020) (see Chapter 2). However, as antibodies' CDR loops are highly variable and distinct from other proteins, this protein-to-antibody 'transfer learning' does not show conclusive benefits (Pittala et al. 2020).

1.3.3 Antibody structure prediction

In this thesis, I develop structure-based prediction tools of antibody properties (see Chapter 2). The usefulness of such tools that require structures as input depends on the ability to accurately predict antibody structures given the lack of readily available crystal data (see Section 1.3.2.2).

1.3.3.1 Homology modelling

Classical antibody structure prediction tools, such as ABodyBuidler (Leem et al. 2016) and FREAD (Choi et al. 2010), use simple homology modelling and database searches to select the best experimental template for a given sequence. These techniques succeed in predicting the six CDR loops with Root Mean Square Deviations (RMSDs) between 0.6Å and 3.5Å, corresponding to the short L2 and longer H3 loops, respectively. Homology methods are limited though given the sparsity of structural data available to use as templates.

1.3.3.2 Deep learning

Recently, deep learning (DL) methods have improved structure prediction, from the advent of AlphaFold (Jumper et al. 2021) to antibody-specific methods such as IgFold (Ruffolo et al. 2023), ABlooper (Abanades et al. 2022), and ABodyBuilder2 (Abanades et al. 2023). ABodyBuilder2 can achieve RMSDs comparable to the resolution of crystal structures for five of the six CDR loops and the framework which can now all be modelled with sub-angstrom accuracy,

leaving only the hypervariable CDRH3 loop as still highly challenging.

Unlike classical homology modelling tools, these DL methods use no templates. Instead, Graph Neural Networks (GNNs, see Section 1.3.4.6) are typically used to predict an antibody's structure from its sequence (Satorras et al. 2021). GNNs work in this instance by first representing an antibody's residues or atoms as nodes in a graph. Edges are defined to exist between nodes close in space. The starting positions of these nodes are arbitrary e.g. they may be randomly positioned, or positioned in a straight line. During the training of the GNN 'messages' are passed between nearby nodes via matrix multiplications (Message Passing Neural Network, MPNN). These messages cause the node positions and features to update, with the objective of matching the true antibody structure.

Though not yet perfect, these advancements in structural modelling mean we can now build ML tools that take as input model structures and output reasonably accurate predictions of antibodies' functions and physical properties, such as their paratope location (Pittala et al. 2020; Chinery et al. 2023) and developability (Raybould et al. 2019). Excitingly, many more ML architectures, besides GNNs, can also be used to predict further antibody properties from their sequences or modelled structures.

1.3.4 Antibody property prediction and design with machine learning

Machine learning can be used to predict antibody properties in many ways. Firstly, the granularity of predictions can vary - antibody properties can be predicted at the residue level e.g. which amino acids belong to the paratope? Alternatively, properties can be determined at a more global level e.g. is this antibody human or does it bind a specific target? Training models to predict these properties can also be performed using various architectures that require different inputs and use either supervised or unsupervised learning approaches (where labels are or are not known).

I cannot cover all machine learning methods currently used for antibody design in this thesis. Instead, I provide an overview of the main methods used by ourselves and others that appear in later chapters, including Decision Trees (DTs), Convolutional Neural Networks (CNNs),

Recurrent Neural Networks (RNNs), Masked Language Models (MLMs), Support Vector Machines (SVMs), and Equivariant Graph Neural Networks (EGNNs). Further details, including architecture specifics and training hyperparameters, are described in later chapters where appropriate.

1.3.4.1 Decision Trees

DT methods (see Figure 1.3.2) are some of the simplest and most commonly used ML methods as their predictions are interpretable (Gilmore et al. 2021). These methods can be applied to aligned, padded, flattened (1D), one-hot encoded antibody sequences. As the name suggests, DTs learn to ask a series of optimised questions with binary answers e.g. is there a Glycine at position 28 in the sequence? Following each question node in the tree lies another question until, finally, a classification is reached e.g. yes, the input heavy chain looks human.

In some instances, multiple DTs may be trained on subsets of the same data and their outputs averaged to reduce overfitting and improve accuracy. These ensemble models are known as Random Forests (RFs). In this thesis we use existing RF methods, such as Hu-mAb (Marks et al. 2021), to classify whether input VH and VL sequences are human (see Chapter 4). In Chapter 3, we also apply our own DT methods to predict whether variants of one antibody therapeutic will maintain high antigen binding affinity based solely on their CDRH3 sequence.

DT methods are quick to implement but the learned trees can become large leading to slow inference. Alternative ML architectures, such as CNNs, can provide smaller but equally accurate networks in some instances by examining multiple residues simultaneously.

1.3.4.2 Convolutional Neural Networks

Like DTs, CNNs can be trained on antibody sequences in a supervised manner to match the ground truth labels. CNNs work by using sliding 'kernels' (matrices) that pass over the aligned, padded 2D sequence vectors of each chain (see Figure 1.3.3). This sliding kernel offers a clear view of multiple residues on either side of each target residue, but no view of others. In this thesis, the kernels used are of the same 'height' as the encoded feature vectors and so traverse the input 'horizontally' only (see Figure 1.3.3), unlike traditional CNN image classification

kernels that also move vertically.

The outputs of each sliding kernel provide residue-level predictions. Antibody-level predictions can be obtained by combining the residue-level outputs using fully connected linear layers, sometimes called multi-layer perceptions (MLPs) (Meyer-Baese et al. 2014). MLPs learn how much weight should be given to previous layer elements in determining the values in the next layer. MLPs can be appended to many ML architectures in this way and can learn better ways to combine residue-level predictions than simply averaging their outputs.

CNNs are used in Chapters 3 & 4 to predict antibody affinity and humanness. Full details on these implementations are provided later. CNNs are also used by Parapred (Liberis et al. 2018), a comparator method, in Chapter 2 in their paratope prediction protocol.

We find that CNNs predict different antibody properties well, even when trained on little data. However, CNNs only explicitly capture local sequence motifs within the width of the applied kernel, especially for residue-level predictions. Alternative methods, such as RNNs and Transformers, better consider the full antibody sequence in these more granular predictions.

1.3.4.3 Recurrent Neural Networks

Similarly to CNNs, RNNs operate on 2D-encoded antibody sequences (Hochreiter et al. 1997). RNN architectures (see Figure 1.3.4), such as Long Short-Term Memory (LSTM) models, are designed to process sequential data - these models were the 'brains' behind some of the earliest language translation tools before transformer architectures, such as ChatGPT (Brown et al. 2020), were invented. For antibodies, RNNs process each chain of amino acids separately and provide predictions for each residue in turn based on their feature vectors and the 'memory' of the residues before it. This memory will accumulate for each chain as more residues are processed but it will be most fresh for amino acids immediately before the residue in question. Bi-directional LSTM models offer improvements over classical RNNs by maintaining better memory of more distant residues and parsing each chain in both directions.

We do not design our own RNN architectures in this thesis but they are used by Parapred (in addition to CNNs) to obtain paratope predictions in Chapter 2.

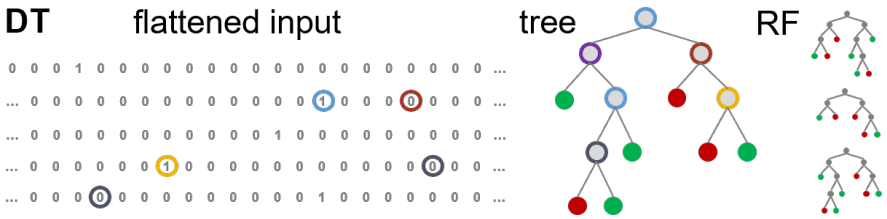

Figure 1.3.2: Decision Tree. A numbered and padded one-hot sequence encoding is first flattened to be 1D in shape. A series of optimised yes-no questions (colour-coded in the tree) are then asked about the input e.g. is there a '1' at index 32 in the input? Questions can repeated at different levels e.g. the blue circles. The final node in each path (green or red) indicates the antibody classification. Random Forests (RFs) involve a collection of DTs. For RFs, the final classification is given by the majority vote of all DTs.

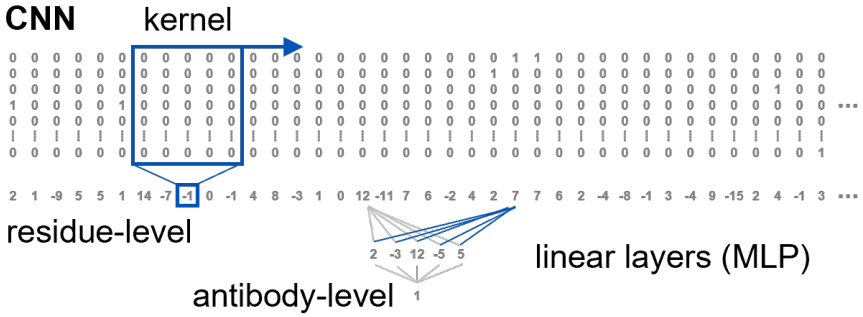

Figure 1.3.3: Convolutional Neural Network. A learned kernel passes over the input 2D encoded sequence, providing residue-level outputs based on the immediate sequence neighbours of each residue. Antibody-level outputs can be obtained from fully connected multi-layer perceptions (MLPs) that combine and weight values from previous layers (only some connections are shown for plotting clarity).

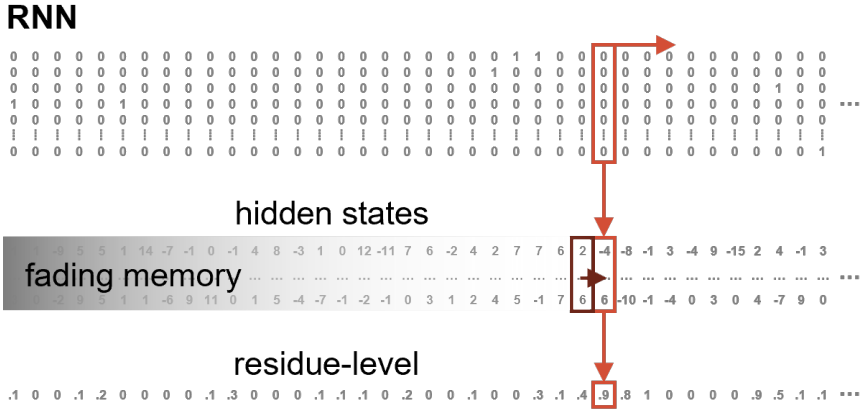

Figure 1.3.4: Recurrent Neural Network. Residue-level outputs are calculated based on the input residue feature vector and a hidden 'memory' of the previous residues in the sequence. The memory of residues at the start of the sequence wanes as more residues are processed.

RNNs offer information from more distant residues than CNNs but inference is slow as sequences must be processed one residue at a time. Furthermore, RNNs do not consider that sequence-distant residues can be close in the 3D antibody structure and should perhaps contribute more to the 'memory'. Transformer-based architectures, such as Masked Language Models, offer improvements on this via their 'attention' mechanism.

1.3.4.4 Masked Language Models

Transformers, first introduced in 2017 (Vaswani et al. 2017), are a recent and powerful addition to the world of machine learning and natural language processing. Unlike RNNs, transformers process data in parallel. Parallel processing can offer speed advantages for long inputs, but more importantly, it allows 'attention' to be used to capture long-range sequence dependencies in the data. For example, attention means networks can hopefully more easily learn that different CDR loops might affect each other most, despite being distant in antibody sequences.

Transformers now underpin most Large Language Models (LLMs) common in science and the wider world (Devlin et al. 2018; Brown et al. 2020; Touvron et al. 2023). Masked Language Models (MLMs, see Figure 1.3.5) are particularly prevalent in the fields of protein and antibody design. MLMs are trained in an unsupervised fashion on large amounts of sequence data. In this process, residues are randomly masked and the MLM is tasked with predicting their identity based on the context of the unmasked residues. For each position, every amino acid type will receive a probability whose scores will sum to one. Novel but sensible antibody variants can then be designed from these probabilities by preferentially selecting amino acids that have high likelihoods. If masking a large stretch of amino acids, such as the entire CDRH3 loop, this process is sometimes referred to as 'one-shot' or 'zero-shot' design as only one antibody template and no original CDRH3 information is used to design variants.

MLMs may be trained on all available protein sequences, or they can be trained on subsets of data to optimise them for certain tasks. For example, MLMs trained entirely on human sequences will learn to give high probabilities to common human residues/patterns of residues at each position. If a mouse sequence is subsequently input and scored by the MLM, many residues will receive low scores. As a result of this, MLMs can humanise sequences by mutating

residues to more 'likely' ones and offer humanness prediction scores by averaging the residue-level MLM outputs.

In this thesis, MLMs are used for both antibody design and property prediction. In Chapter 3, we use masked residue likelihoods from ESM (Rives et al. 2021) and AbLang (Olsen et al. 2022a) to design high-affinity variants of an antibody therapeutic. Later, in Chapter 4, we use an alternative MLM - Sapiens (Prihoda et al. 2022) - to score and humanise a larger set of antibody therapeutic precursor sequences.

Transformer-based LLMs have the potential to learn long-range structure dependencies from sequences alone. However, explicitly structure-base ML architectures also exist that may offer accuracy improvements in certain areas, despite the sparsity of structural training data.

1.3.4.5 Support Vector Machines

A 3D antibody structure can be represented as voxels (3D pixels) or a graph. Once transformed to either 3D representation, SVMs can be used to learn a 'hyperplane' separating voxels or nodes belonging to different classes (see Figure 1.3.6).

We do not use SVMs in this thesis, but in Chapter 2 we compare our method to one from Daberdaku *et al.* (Daberdaku et al. 2019). This work used a voxel antibody representation and an SVM to separate voxels containing paratope residues from non-paratope voxels.

Voxel representations and SVMs address some issues of using sequence alone but accuracy improvements are still possible. Therefore, in this thesis, we use EGNNs, similar to those used by antibody structure prediction tools to predict both residue-level and antibody-level properties.

1.3.4.6 Equivariant Graph Neural Networks

EGNNs (Satorras et al. 2021), used synonymously with Message Passing Neural Networks (MPNNs) in this thesis, can be configured to update both node features and coordinates for structure prediction tasks (see Section 1.3.3.2). For property prediction tasks, node coordinates can remain fixed. In these instances, messages simply pass between nodes close in space (see

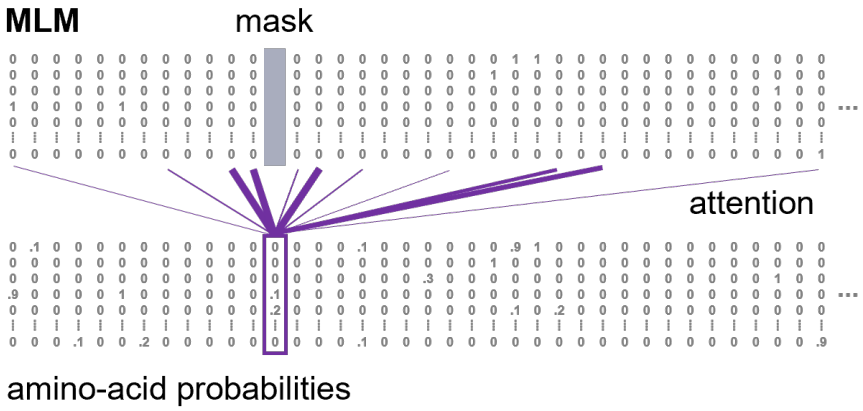

Figure 1.3.5: Masked Language Model. MLMs are built using the transformer architecture and can be trained in an unsupervised fashion. MLMs have visibility of all residues in the sequence at once and learn which to pay the most 'attention' to (thicker lines) when calculating the amino acid probabilities of the masked residue(s).

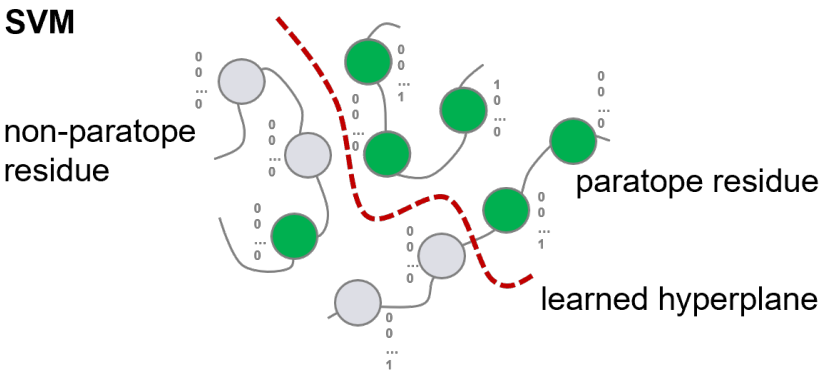

Figure 1.3.6: Support Vector Machine. SVMs can act on 3D representations of antibodies (graphs or voxels) though the figure shows a simplified 2D view of only ten residues from three CDR loops (grey lines). SVMs learn a hyperplane (red dashed line) that aims to separate residues belonging to two classes e.g. paratope and non-paratope residues. The learned hyperplane does not always perfectly separate the two classes e.g. one paratope residue (green) is separated from the rest.

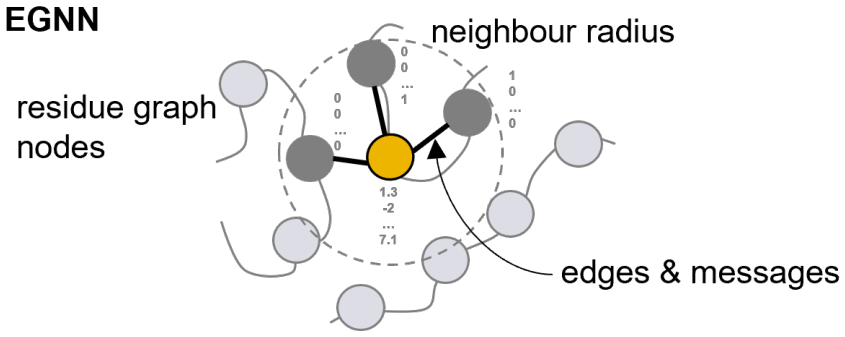

Figure 1.3.7: Equivariant Graph Neural Network, synonymous to MPNNs (Message Passing Neural Networks) in this thesis. EGNNs update the feature vectors of each graph node (residue) based on the context of their surrounding neighbours via 'messages' passed along 'edges'. If multiple EGNN layers are used, messages can propagate further in the graph, allowing residues to affect others outside their immediate neighbour radius.

Figure 1.3.7). If multiple EGNN layers are used, messages can propagate further in the graph allowing residues to affect others outside their immediate neighbour radius. Combined with MLPs, EGNNs can offer either residue-level predictions, such as paratope prediction (Vecchio et al. 2021), or antibody-level predictions, such as binding affinity (Hummer et al. 2023).

Additionally, GNNs can be used for antibody design using 'inverse folding' (IF) (Dauparas et al. 2022; Høie et al. 2024). IF aims to solve the opposite challenge of antibody structure prediction i.e. given, an antibody backbone structure, which amino acids are most likely to fold into these positions. GNNs underpin IF methods similarly to structure prediction methods. Like MLMs, IF methods can be applied to antibody design by masking the amino acid labels of all residues in a CDR loop, for example. The IF method then returns amino acid likelihoods for each position and the top-ranked amino acids can then be selected or the likelihoods can be used as sample weights to design many different sequences from a single starting sequence.

EGNNs appear in the first two chapters of this thesis. In Chapter 2 we present our novel method, Paragraph (Chinery et al. 2023), for paratope prediction and compare this to an alternative graph-based method, PECAN (Pittala et al. 2020). In Chapter 3 we use the IF tool ProteinMPNN (Dauparas et al. 2022), in addition to MLMs, to design high-affinity antibody variants. Finally, in Chapter 3, we also apply an EGNN adapted from Hummer *et al.* (Hummer et al. 2023) to classify high-affinity antibody variants.

1.4 Thesis overview

The previous sections have described some of the challenges of developing antibody therapeutics and several ML methods that may assist in tackling these. Below, we summarise the specific steps taken in this thesis to address the problems of paratope prediction, affinity maturation, and humanisation.

In Chapter 2, we describe our novel paratope prediction tool, Paragraph. Paragraph uses an EGNN with simple feature vectors to offer state-of-the-art paratope prediction in a tenth of a second per structure. These rapid, accurate predictions offer potential improvements to computational docking experiments and the guidance of optimising mutations.

In Chapter 3, we explore affinity optimising mutations for Trastuzumab, an antibody where the paratope is already known, with the aim of baselining deep learning affinity maturation techniques. Recently, excitement has grown in this area of protein design in particular. To provide context to this growth, we apply existing classical and ML methods to classify high-affinity binders from a new, large dataset of affinity-labelled antibody variants. We then compare the enrichments of novel libraries designed using five different computational approaches. Finally, we simulate how continuous, adaptive learning methods might be integrated with wet-lab experiments to refine antibody libraries as experimental data becomes available.

In Chapter 4, we adapt our simple ML architecture introduced in the previous chapter to develop Humatch, an antibody humanisation tool that suggests optimising mutations that align highly with experiments. Humatch consists of three CNN classifiers. The first two classify whether an antibody VH and VL are human with near-perfect accuracy. The third CNN classifies whether the human VH and VL are well-paired, indicative of overall stability. During humanisation, Humatch jointly pushes VH and VL sequences to achieve high scores with all CNNs. Humatch's humanisation protocol ensures designs do not get stuck in local minima and also guides each chain towards target V-genes and away from others.

Lastly, we conclude our work and suggest future directions that could be pursued within the areas of binding site prediction, affinity maturation, and humanisation.

2 | Antibody paratope prediction using simple, structure-based deep learning methods

2.1 Abstract

This chapter introduces 'Paragraph', a structure-based paratope (antibody binding site) prediction tool that outperforms current state-of-the-art methods using simpler feature vectors and no antigen information. Knowledge of the paratope is useful in computational drug discovery as it can influence where optimising mutations are made. For example, affinity-improving mutations should largely focus on paratope residues, while humanising mutations that seek to reduce immunogenicity but retain binding should avoid the paratope. Paratope information can also help guide computational docking experiments, resulting in faster and more accurate predicted antibody-antigen complexes.

The following text is adapted from *Paragraph - antibody paratope prediction using graph neural networks with minimal feature vectors* (Chinery et al. 2023).

2.2 Introduction

As described in Chapter 1, antibodies can bind to and neutralise their target antigens with high specificity and affinity. Their high specificity and low immunogenicity mean antibodies have grown to dominate the therapeutic marketplace (Lu et al. 2020) (see Section 1.2.1). Current experimental methods to determine how an antibody and antigen bind, and hence the potential neutralising effect, are slow and costly (Wouters et al. 2020). Computational docking tools (Pierce et al. 2014; Zundert et al. 2016; Eberhardt et al. 2021) have been developed to predict binding, offering a cheaper, faster alternative. However, current computational tools often fail to accurately recapitulate antibody-antigen binding (Ambrosetti et al. 2020a) and are generally not feasible for use in a truly high throughput fashion (Bender et al. 2021).

Predicting the binding site of an antibody can both allow better estimation of its bound pose (Ambrosetti et al. 2020b) and indicate key residues to mutate to change binding properties (Rabia et al. 2018) (see Section 1.2.4.1). Parapred (Liberis et al. 2018) is a widely used and freely available sequence-based paratope prediction tool that uses convolutional and recurrent neural networks (see Sections 1.3.4.2 & 1.3.4.3). More recent attempts at paratope prediction e.g. PECAN (Pittala et al. 2020), have brought in structural information, representing the antibody structure as a graph (see Section 1.3.4.6). PECAN requires both antibody and antigen structures as input into a graph convolution attention network, but based on that input outperforms Parapred.

In this chapter, we use methods that can accurately and rapidly predict 3D antibody structures (Leem et al. 2016; Abanades et al. 2022) (see Sections 1.3.3.1 & 1.3.3.2) to build a structure-based paratope predictor, Paragraph. Paragraph assumes antibodies are highly specific to only one epitope (see Section 1.1.4.2) so does not condition its predictions on any antigen information. Additionally, as crystal structures will only be available for a minority of antibodies of interest (see Section 1.3.2.2), structure-based paratope predictors must work on predicted 3D model structures. Paragraph takes as input the modelled structure of an antibody and, using equivariant graph neural network layers (Satorras et al. 2021), rapidly predicts the probability of residues belonging to the paratope. Paragraph outperforms the state-of-the-art paratope

predictor, PECAN, when trained on the same dataset.

2.3 Materials and Methods

2.3.1 Train-test dataset creation

2.3.1.1 Existing datasets

Paragraph was trained and tested on the same complexes used by PECAN (Pittala et al. 2020). This dataset contains 460 antibody-antigen complexes - 205 complexes are used for training, 103 for validation, and 152 for testing. PECAN's dataset is a subset of the 472-complex dataset compiled by Daberdaku et al. (Daberdaku et al. 2019) (see Section 1.3.4.5). The reduced dataset contains only complexes with paired heavy and light chains, sub 3Å resolution, and protein or peptide antigens. CSVs containing the PDB codes and heavy, light and antigen chain IDs used in our training, validation and testing can be found at www.github.com/oxpig/Paragraph.

In our research, we identified issues with certain structures in the PECAN dataset - a subset of these are shown below. These issues, along with the large growth in structural data over the past three years, inspired us to create our Expanded dataset (see Section 2.3.1.2). The 'problem structures' we identified were left in our training, validation, and test sets when using PECAN's data for a fair comparison against existing methods. However, steps were taken when compiling our Expanded dataset to try to remove such structures.

1IGC - antigen not binding F_v

PDB structure 1IGC is found in the PECAN train set and includes an antigen bound to the constant region of the F_{ab}. No contacts exist in the F_v region (see Figure 2.3.1), the only region searched for paratope residues.

4ERS - antigen misplaced in PDB file

PDB structure 4ERS is found in the PECAN validation set. On the RCSB webpage, the true binding confirmation of antibody and antigen can be found. However, in the PDB file, the incorrect antigen has been paired with the antibody (see Figure 2.3.2).

1TZI - One chain of antigen homodimer removed in PDB file

PDB structure 1TZI is found in the PECAN validation set. On the RCSB webpage, it is clear that the antibody is binding both chains of the homodimer antigen. However, in the PDB file the dominantly binding chain of the homodimer has been removed (see Figure 2.3.3).

2.3.1.2 Expanded dataset

To take advantage of the ever-increasing amount of structural data now available, we also trained Paragraph on a larger dataset. This new dataset was extracted from the Structural Antibody Database (SAbDab) (Dunbar et al. 2014; Schneider et al. 2022) on 31/03/2022 and includes 1,086 complexes which were divided into train, validation, and test sets using a 60-20-20 split. To generate this new dataset we included only paired heavy and light chain X-ray complexes with 3Å resolution or below. We allowed both protein and peptide antigens, as defined by SAbDab.

To overcome data quality issues observed in the PECAN dataset we required that the F_v contained 10 or more binding residues and that 50% or more of these binding residues were in the Complementarity Determining Region (CDR) loops plus two residues on either side (CDR±2). These two requirements removed just 5% of the dataset.

Finally, using CD-HIT (Fu et al. 2012), we ensured no two antibodies shared over 95% heavy and light chain sequence identity and we removed any structures ABodyBuilder (Leem et al. 2016) could not model using a 95% identity threshold (where no suitable templates were found for homology modelling of the framework region, see Section 2.3.3).

Entries of the Expanded dataset can be found at www.github.com/oxpig/Paragraph.

2.3.1.3 Paratope definition

Following previous methods (Liberis et al. 2018; Pittala et al. 2020), antibody residues were labelled as belonging to the paratope if any heavy atom (any atom that is not Hydrogen) was within 4.5Å of an antigen heavy atom. These labels were obtained by examining crystal structures and the same labels were applied to modelled structures of the same antibodies.

Figure 2.3.1: A cartoon representation of 1IGC taken from https://www.rcsb.org/. The heavy chain is shown in orange, the light chain in green, and the antigen in purple. The antigen binds entirely outside of the F_v region.

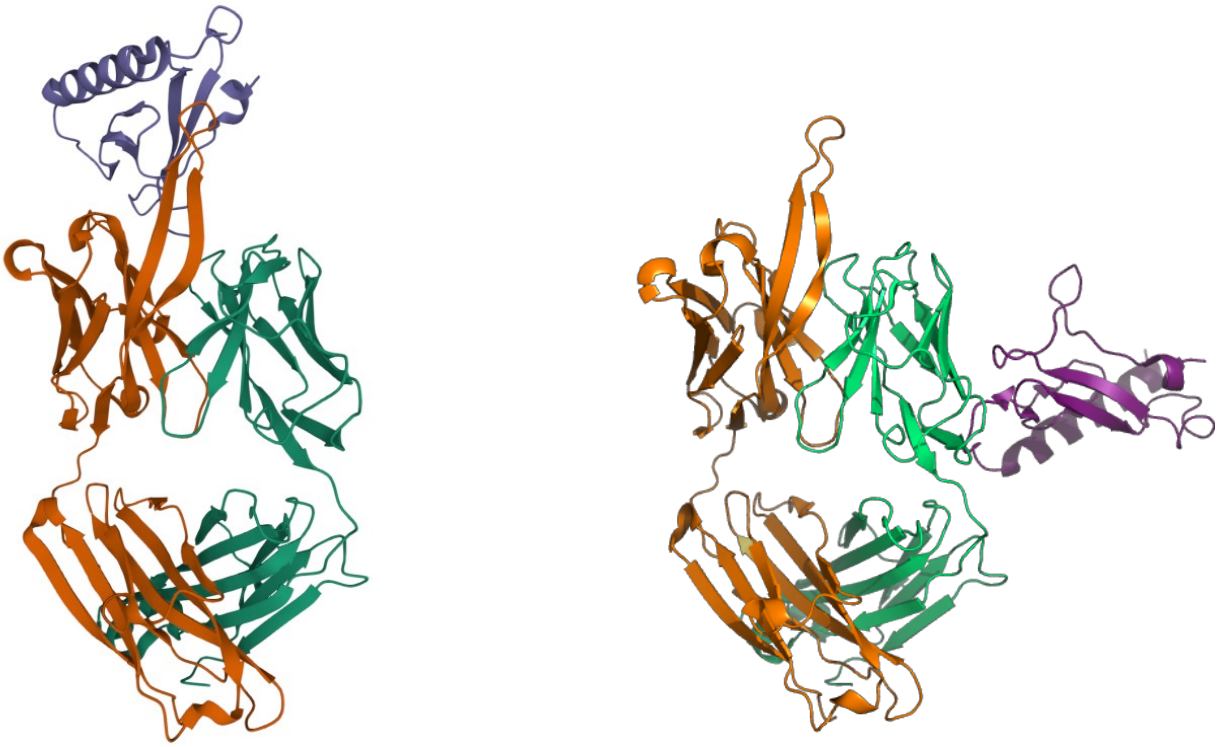

Figure 2.3.2: Left image shows a cartoon representation of 4ERS taken from https://www.rcsb.org/. The heavy chain is shown in orange, the light chain in green, and the antigen in purple. The right image was created in PyMOL where colours have been chosen to approximately match the RCSB image. The antigen appears in two different positions in the two figures.

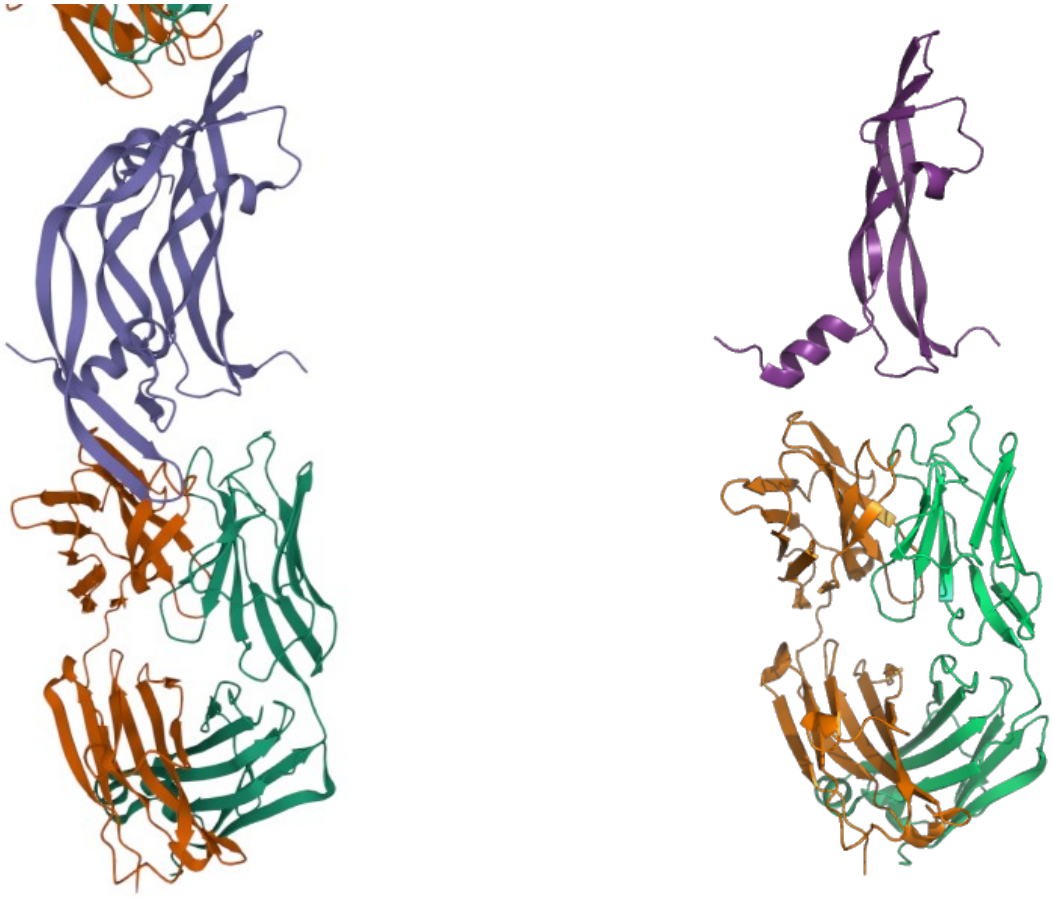

Figure 2.3.3: Left image shows a cartoon representation of 1TZI taken from https://www.rcsb.org/. The heavy chain is shown in orange, the light chain in green, and the antigen in purple. The right image was created in PyMOL where colours have been chosen to approximately match the RCSB image. The antigen chain that dominates binding in the left figure is not present in the PDB file.

2.3.2 Baseline frequency predictions

In addition to comparing Paragraph to existing paratope prediction tools, we also constructed a simple baseline study. The baseline measured the proportion of residues found to bind the antigen in our training set at each IMGT (Lefranc et al. 2003) sequence position. We then predicted this same proportion for every corresponding position in our test set.

This baseline study is important to determine whether or not machine learning methods are learning beyond simple observations and to test the usefulness of different evaluation metrics for the task at hand.

2.3.3 Antibody structural modelling

As crystal structures will only be available for a tiny number of antibodies of interest (see Section 1.3.2.2), it is essential for structure-based paratope predictors to work well on predicted 3D model structures. To maximise the performance of Paragraph on model structures, we trained on a combination of both crystal and model structures and validated on model structures only.

To build our final models, we used ABodyBuilder (Leem et al. 2016) (see Section 1.3.3.1) to model the highly-conserved framework region and ABlooper (Abanades et al. 2022) (see Section 1.3.3.2) to model the hyper-variable CDR loops, not allowing the use of sequence identical templates.

2.3.4 EGNN classifier

2.3.4.1 Architecture

As mentioned in Section 2.3.3, we trained Paragraph on a combination of crystal and model structures and validated on model structures only.

For both crystal and model structures, we generated a graph representation of the antibodies by describing each residue in the CDR±2 as a single node (see Figure 2.3.4). The nodes were

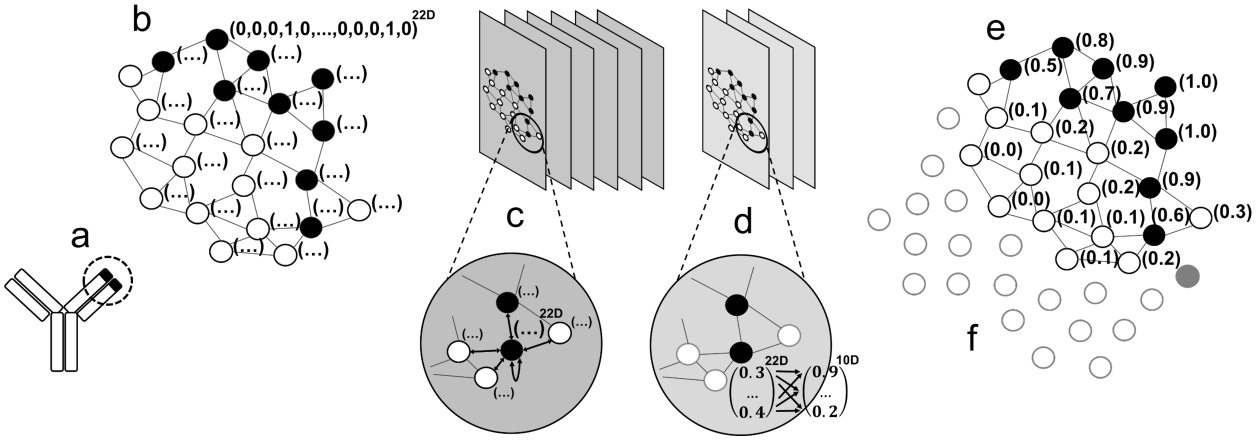

Figure 2.3.4: An overview of Paragraph's architecture. **(a)** As the paratope is predominantly found in the CDR loops (six loops on the end of each antibody 'arm'), only residues in the CDR loops plus two extra residues on either side (CDR±2) are represented as nodes. Residues labelled as belonging to the paratope are shown as filled circles throughout. Non-paratope residues are shown as outlines only. **(b)** Node feature vectors are one-hot encodings of the amino acid type (20D) and chain type (2D). Edges exist between nodes separated by ≤10Å. **(c)** The graph representation of the antibody is passed through a series of six equivariant graph neural network layers. This allows residues to learn about other antibody residues close to them in space, but not necessarily in sequence. **(d)** The 22D node feature vectors are passed through three linear layers with output dimensions 10, 10, and 1. **(e)** Paragraph outputs a binding probability between zero and one for each residue in the CDR±2. **(f)** We extend our results over the entire F_v region by predicting zero for all other residues (light grey).

$$\mathbf{m}_{ij} = \phi_e\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \|\mathbf{x}_i^l - \mathbf{x}_j^l\|^2, a_{ij}\right) \tag{2.1}$$

$$\mathbf{m}_i = \sum_{j \neq i} \mathbf{m}_{ij} \tag{2.2}$$

$$\mathbf{h}_i^{l+1} = \phi_h\left(\mathbf{h}_i^l, \mathbf{m}_i\right) \tag{2.3}$$

Figure 2.3.5: Equations from Satorras *et al.* describing how node features and positions are updated at each step of an Equivariant Graph Neural Network. $\mathbf{m}_{ij}$ describes the message passed between nodes i and j at a given layer. $\mathbf{h}_i^l$ describes the feature vector of node i at layer l, and $\mathbf{x}_i^l$ describes the node's co-ordinates at that layer. We initiate $\mathbf{h}_i^0$ as 22D vectors comprised of one-hot encodings of the amino acid type and chain type for all residues i. a_{ij} is the edge feature between nodes i and j. In our network, this is simply an adjacency matrix defined by a 10Å distance cut-off between antibody residues. ϕ are the non-linear node (h) and edge (e) operations used to update the various graph attributes. In our network, we leave the coordinates of our nodes fixed.

defined to have coordinates given by the C_α atoms. Directionless edges were defined to exist between nodes separated by 10Å or less. Each node was assigned a 22D feature vector comprised of a 20D one-hot encoding of the amino acid type and a 2D one-hot encoding of the chain type. Following previous methods (Liberis et al. 2018; Pittala et al. 2020), residues were labelled as binding if any heavy atom was within 4.5Å of an antigen heavy atom.

To predict the binding of each residue, a series of six graph and three linear layers were used. The graph layers were adapted from LucidRain's (LucidRains 2021) implementation of the architecture described in *E(n) Equivariant Graph Neural Networks* (EGNN) (Satorras et al. 2021). Figure 2.3.5 shows the key equations that define an EGNN layer and further background can be found in Section 1.3.4.6. In our network, the coordinates of our nodes are fixed.

All graph layers have input and output feature dimensions of 22 and use skip connections. The linear layers have output dimensions of 10, 10, and 1. Following each graph layer the non-linear Hardtanh activation function is applied. Hardtanh activation is also applied following the first two linear layers, while a Sigmoid function is applied to the output of the final layer to yield a probability between zero and one.

2.3.4.2 Training

Paragraph was trained for 16 different random seeds, each for 300 epochs. Batch sizes of one were used as graph sizes varied due to differing CDR loop lengths. Gradients were accumulated across 16 batches. The Binary Cross Entropy (BCE) loss was calculated between the residue labels and predictions. This loss was then optimised using Adam stochastic gradient descent with a learning rate of 0.001. An imbalance weighting of three was used to account for the class imbalance between residues classed as binding the antigen (27%) and those classed as not binding (73%). For each seed, weights were saved that delivered the lowest BCE loss on the validation dataset. The weights that resulted in the largest area under the precision-recall curve (PR AUC) for the validation set were then used to predict the paratope for models of our test set.

To extend our predictions over the entire F_v we predicted zero for all residues outside the CDR±2. This simple extension works well as only 1% of residues outside the CDR±2 are labelled as binding in both PECAN's and our Expanded datasets. We used this approach instead of training over and predicting residues in the entire F_v region as the large class imbalance (10:1) resulted in high instability.

2.4 Results

In the following sections, we show Paragraph offers more accurate paratope predictions than existing methods. We also break these predictions down by CDR loops and by the quality of the input antibody structure. In all sub-studies, we include baseline performances to confirm we are learning deeper patterns than position frequencies alone. Finally, we present a qualitative study that demonstrates Paragraph can determine when a paratope is not centred in the middle of the CDR loops.

2.4.1 Paragraph achieves higher PR AUC than existing methods

The top row of Table 2.4.1 shows the results of a simple baseline that looked at how often each IMGT position was found to bind the antigen in our training data (see Section 2.3.2). This baseline achieved the highest area under the receiver operating characteristic curve (ROC AUC) of all methods. This behaviour was expected given the uneven distribution of positive (paratope) and negative (non-paratope) residues throughout the F_v, allowing many negative residues to be easily and accurately classified based on frequency alone (Richardson et al. 2024). This baseline study demonstrates the importance of considering the area under the precision-recall curve (PR AUC) for this particular class-imbalanced problem.

Method	PR AUC	ROC AUC
Baseline	0.626	**0.952**
Daberdaku *et al.* (Daberdaku et al. 2019)	0.545	0.923
Parapred (Liberis et al. 2018)	0.646	0.930
PECAN (Pittala et al. 2020)	0.675	**0.952**
Paragraph	**0.696**	0.934
Paragraph (Expanded dataset)	***0.725***	***0.934***

Table 2.4.1: Comparison of paratope prediction methods. All are evaluated on model structures over the F_v region. All results use PECAN's dataset unless stated otherwise. Daberdaku et al.'s dataset includes 12 additional complexes with non-protein antigens. Daberdaku et al. and PECAN's performance have been taken from the original papers (PECAN's numbers are approximated from the figures provided). We retrained Parapred using PECAN's dataset and extended the results over the F_v by predicting zero for all additional residues, similar to our method. Our baseline shows the importance of considering PR AUC in this unevenly distributed, class-imbalanced problem.

Using true paratope labels, Paragraph outperformed both PECAN (Pittala et al. 2020) and Parapred (Liberis et al. 2018), the best performing freely available methods (see Table 2.4.1). Parapred was retrained and tested for our analysis, but we were unable to re-evaluate PECAN on our newer model structures or retrain it on our Expanded dataset due to a lack of easily reproducible code.

Using our newer ABodyBuilder+ABlooper models and evaluating Paragraph on 1,000 bootstrapped samples of PECAN's test set, Paragraph achieved a mean PR AUC of 0.695±0.015 (see Section 2.4.4 and Figure 2.4.2). The standard deviation of our validation set PR AUCs for all 16 different training seeds was 0.008. Paragraph's performance increased further when using the larger dataset of structures now available (see Table 2.4.1).

Finally, Paragraph requires no antigen information, meaning it can be deployed in wider use cases than PECAN, which also requires significant feature engineering. Antigen-agnostic paratope prediction tools, such as Paragraph, are useful and feasible as antibodies are highly specific, very often binding only one epitope with high affinity (see Section 1.1.4.2).

2.4.2 Selecting Paragraph's optimal classifier cut-off

To recapitulate the true number of binding residues observed in the CDR±2 region, we recommend using a classifier cut-off of 0.734 on Paragraph's predictions. This cut-off value was optimised using our Expanded validation dataset. In this dataset, on average we observe 74.0 residues belonging to the CDR±2 region, and 20.2 of these residues belonging to the paratope. A classifier cut-off of 0.734 on Paragraph's predictions reproduces this observed ratio of paratope to non-paratope residues. Using this cut-off value, Paragraph achieves a recall of 0.677 and a precision of 0.705 on our Expanded test set (see Figure 2.4.1).

2.4.3 Paragraph evaluation runtime

Given an existing crystal or model structure, Paragraph is faster than existing paratope prediction methods, taking approximately 0.1s (50 times faster than Parapred, though Parapred can be run on antibody sequences alone).

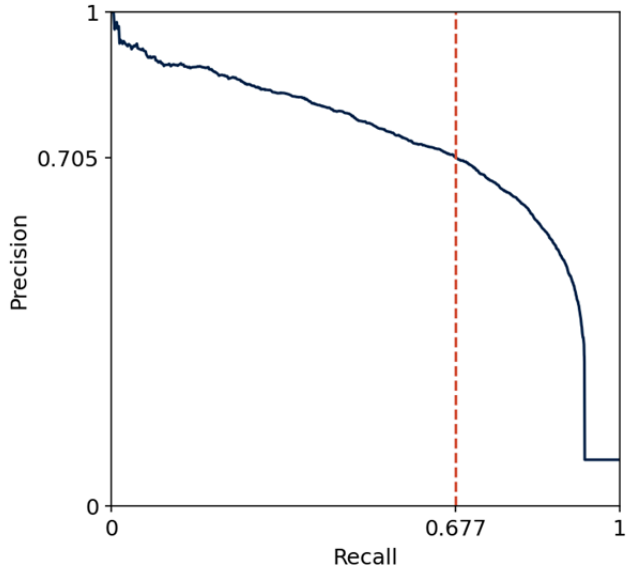

Figure 2.4.1: Precision-recall curve for Paragraph when trained on our Expanded dataset and evaluated on model structures with results extended over the F_v. Due to 8% of paratope residues lying outside the CDR±2, the highest recall Paragraph can achieve is 92%. The horizontal line at precision ∼0.1 after this point shows the 10:1 class imbalance of non-binding to binding residues observed in the F_v. The vertical red line intersecting the PR curve shows the recall (0.677) and precision (0.705) achieved when selecting a classifier cut-off (0.734) that recapitulates the true proportion of binding residues observed in the CDR±2 region.

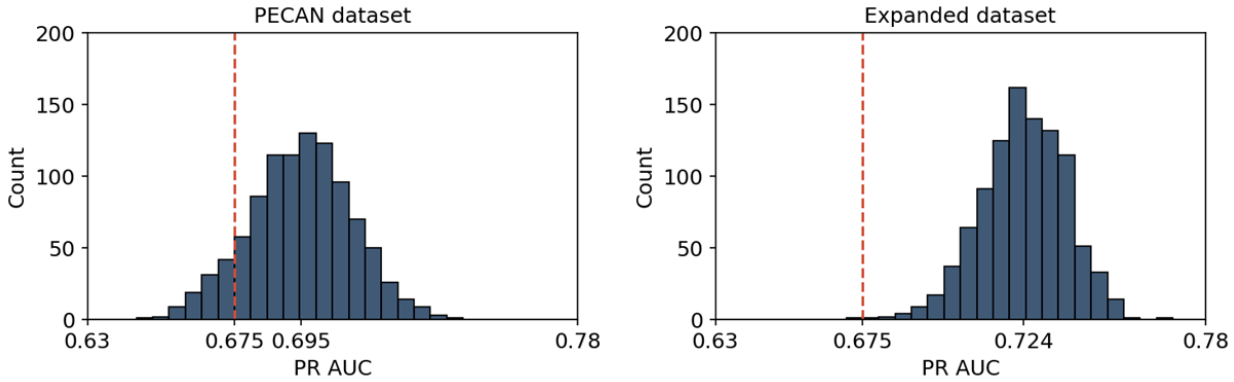

Figure 2.4.2: Performance of Paragraph on 1,000 bootstrapped samples taken from PECAN's (left) and our Expanded (right) test sets. On PECAN's test set, Paragraph achieved a mean PR AUC of 0.695 with a standard deviation of 0.015. PECAN's reported PR AUC of 0.675 is shown in red for comparison. On our Expanded test set, Paragraph achieved a mean PR AUC of 0.724 with a standard deviation of 0.013.

2.4.4 Bootstrapped uncertainty estimation

1,000 bootstrapped samples of PECAN's test set were used to calculate the statistical significance of Paragraph's results. Using PR AUC, we found that Paragraph outperformed PECAN with a significance of 1.4 standard deviations and exceeded PECAN's performance in 90.9% of all samples (see Figure 2.4.2).

Similar calculations were performed using 1,000 bootstrapped samples of our Expanded test set. Likely due to the larger amount of training data, Paragraph exceeded PECAN's reported PR AUC for 999 of 1,000 samples (see Figure 2.4.2). The standard deviation of our bootstrapped samples for our Expanded dataset is smaller than that observed for the PECAN dataset. This narrower distribution may be due to the larger, cleaner dataset used.

2.4.5 Paragraph's accuracy increases with structural model quality

Until recently, structure-based paratope prediction methods were limited in their usefulness by the lack of experimental structural data or accurate models. However, recent advancements in antibody modelling (Leem et al. 2016; Abanades et al. 2022) mean structure-based paratope prediction methods can now be used widely.

Paragraph was initially developed in 2022 and we selected two of the best antibody modelling tools available to us at the time - ABodyBuilder and ABlooper. We chose ABlooper as our preferred CDR modelling tool as it rapidly produces high-quality models and provides a model confidence score. ABlooper's model confidence score is derived by training five separate networks to predict the antibody structure, each initialised with different seeds. When predicting the antibody structure, predictions are obtained by all five networks and the average structure is taken. The Root Mean Square Deviation (RMSD) between the five decoys for each of the CDR loops is evaluated which acts as a proxy for the model confidence. Five of six CDR loops are often predicted with high accuracy. However, the accuracy of the longer, highly diverse CDRH3 loop is more variable and typically far worse than the other loops. If the CDRH3 decoy diversity is high (i.e. the five models vary greatly in structure) then we have low model confidence and vice versa.

Evaluated on	PR AUC	ROC AUC	F-score	MCC
Crystals (all)	0.757	0.937	0.719	0.692
Baseline	*0.624*	*0.952*	*0.654*	*0.622*
Models (all)	0.725	0.934	0.696	0.669
Baseline	*0.624*	*0.952*	*0.654*	*0.622*
Models (top 75%)	0.742	0.937	0.713	0.687
Baseline	*0.656*	*0.956*	*0.670*	*0.637*
Models (top 50%)	0.763	0.939	0.727	0.700
Baseline	*0.671*	*0.957*	*0.676*	*0.643*
Models (top 25%)	**0.800**	0.947	**0.755**	**0.731**
Baseline	*0.723*	***0.964***	*0.695*	*0.662*

Table 2.4.2: Performance of Paragraph across the entire F_v on both crystal and model structures using our Expanded dataset. Baseline performances on all data subsets are included in italics for completeness.

CDR/FR region	PR AUC	ROC AUC	F-score	MCC
CDRL1	0.762	0.857	0.678	0.499
Baseline	*0.691*	*0.799*	*0.642*	*0.435*
CDRL2	0.675	0.876	0.654	0.559
Baseline	*0.238*	*0.741*	*0.555*	*0.429*
CDRL3	0.770	**0.884**	0.747	**0.598**
Baseline	*0.682*	*0.827*	*0.716*	*0.545*
CDRH1	0.735	0.856	0.678	0.515
Baseline	*0.514*	*0.810*	*0.650*	*0.468*
CDRH2	0.789	0.854	0.727	0.498
Baseline	*0.643*	*0.745*	*0.667*	*0.368*
CDRH3	**0.796**	0.866	**0.762**	0.571
Baseline	*0.700*	*0.813*	*0.735*	*0.519*
Framework	0.429	0.768	0.505	0.500
Baseline	*0.410*	*0.952*	*0.436*	*0.427*

Table 2.4.3: Performance of Paragraph broken down over individual CDR loops and the framework region. Results are across the entire F_v on model structures using our Expanded dataset. Baseline performances on all data subsets are included in italics for completeness.

Table 2.4.2 shows that if we select only ABlooper's most confident models (those with the lowest CDRH3 decoy diversity, known to correlate with shorter loop lengths (Abanades et al. 2022)), we can match the performance observed when evaluating Paragraph on crystal structures. All results are provided for our Expanded test set and are evaluated over the entire F_v region. In agreement with Parapred (Liberis et al. 2018), thresholds for calculating the F-score and Matthews correlation coefficient (MCC) were obtained by maximising Youden's index (Youden 1950).

Baseline performances on each data subset have been included for completeness. Our baseline calculation is sequence-based, not structure-based, and so the baseline performance on the 'Crystals (all)' and 'Models (all)' datasets is the same. Like Paragraph, our baseline performance improves with model quality. One reason for this joint correlation could be that shorter CDR loops tend to be modelled more accurately, and they will contain fewer 'rare' sequence positions that our baseline has seen fewer examples of.

2.4.6 Paragraph achieves high accuracy across all CDR loops

Table 2.4.3 shows a breakdown of Paragraph's performance when trained and tested on paired data from our Expanded dataset. All results are provided on ABodyBuilder+ABlooper model structures of our test set and are evaluated over the entire F_v region.

Paragraph achieved high accuracy predictions across all CDR loops, always exceeding baseline PR AUC, ROC AUC, F-score, and MCC metric scores. The CDRH3 and CDRL3 loops that contain the most paratope residues achieved the highest scores, with PR AUCs up to 0.796.

Paragraph's performance over the framework region is lower than the CDR loops due to the larger class imbalance (very few framework residues belong to the paratope) and the fact that Paragraph is restricted to predicting the CDR±2 region only.

Baseline performances on each data subset have been included for completeness. Similar to Paragraph, all baseline metrics, except ROC AUC, are highest for the regions with the least severe class imbalance.

2.4.7 Qualitative study - F_{ab} fragment in complex with CD9 large extracellular loop protein

Figure 2.4.3 shows Paragraph's predictions for the PDB structure 6RLO compared to the ground truth. 6RLO includes the structure of the CD9 large extracellular loop protein bound by a neutralising antibody. 6RLO was chosen for this study to examine whether Paragraph can accurately predict the paratope when the antigen is bound away from the centre of the antibody. The figure shows how Paragraph successfully predicts the highest probabilities for the residues facing the antigen.

This example also highlights a limitation of our approach of predicting only residues within the CDR±2 and assigning zero probability to all other residues. The top two structures in Figure 2.4.3 show that Paragraph 'predicts' zero for two residues outside the CDR±2 that do belong to the paratope in this instance.

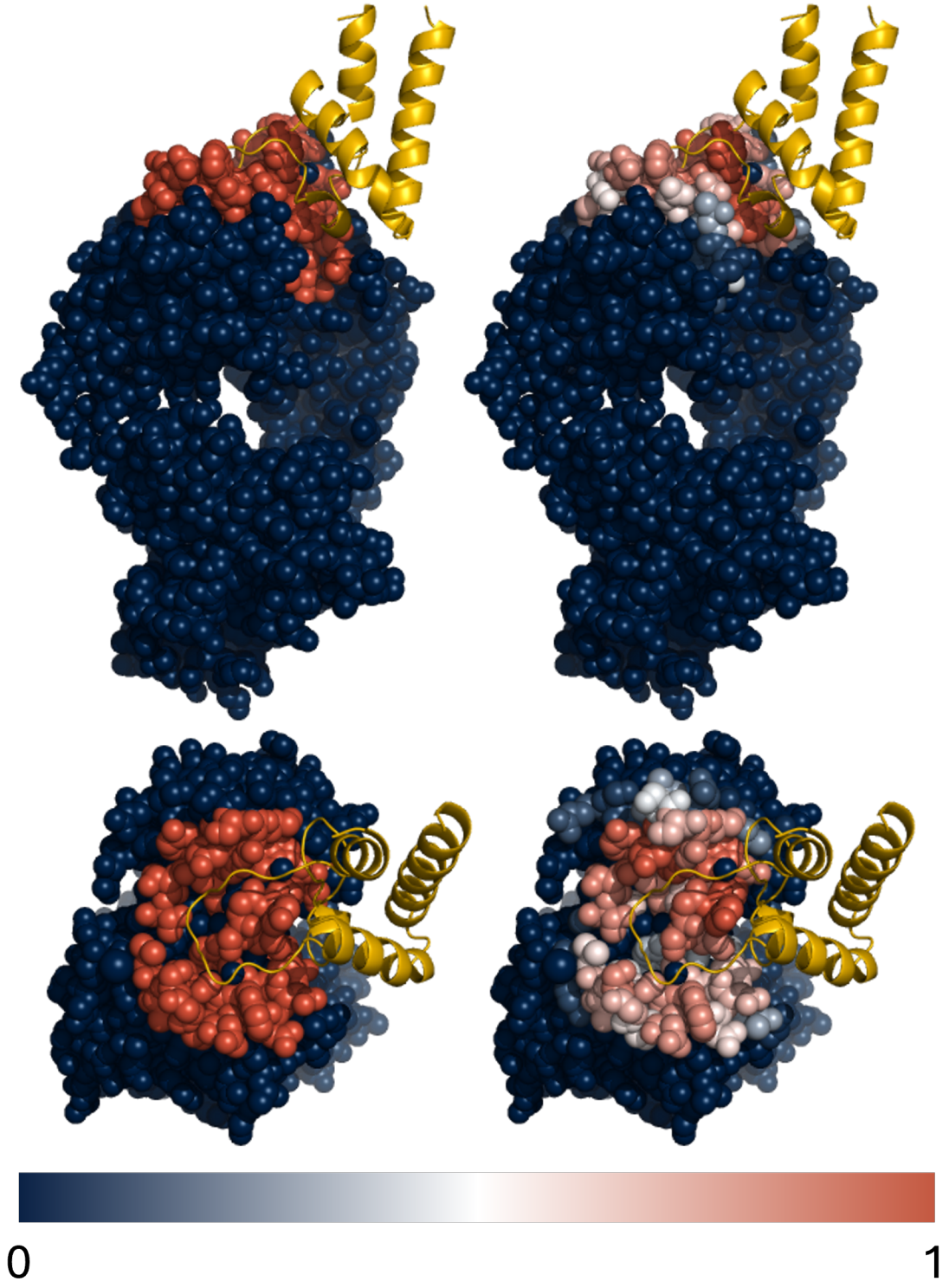

Figure 2.4.3: Side and top views of the F_{ab} region of the CD9 large extracellular loop binding antibody, 6RLO. The two left figures show the ground truth paratope residues in red (residues within 4.5Å of an antigen heavy atom). Non-paratope residues are shown in dark blue. The antigen is shown as a cartoon representation in gold. The two right figures show Paragraph's predicted paratope using a blue-white-red spectrum across the scores from Paragraph. The scores align with the colour bar at the bottom of the figure. Paragraph was trained on our Expanded dataset.

2.5 Discussion

Using equivariant graph neural networks, simple feature vectors, and no antigen information, Paragraph achieves paratope prediction performance above the current state-of-the-art, PECAN, which requires structures of both the antibody and antigen. On the PECAN dataset, Paragraph achieves a PR AUC of 0.696 and the performance increases further when using our Expanded dataset, suggesting as more data becomes available, further improvements may be possible.

Due to the uneven distribution of paratope residues and severe imbalance across the F_v region, care must be taken when selecting evaluation metrics as some metrics may give misleadingly high performance. We show that a simple baseline achieves the highest ROC AUC (0.952) of all paratope prediction methods tested, highlighting the need to use PR AUC when comparing different tools in this instance.

We also stress the importance of using model, not crystal, structures when measuring performance due to the lack of readily available crystal structures in research settings. Were crystal structures to be used for testing in place of model structures, Paragraph's PR AUC on our Expanded dataset would increase from 0.725 to 0.757.

Similarly, when considering only ABlooper's 50% most confident models, Paragraph achieves a higher PR AUC of 0.763 on our Expanded dataset. Further advancements in antibody structure prediction should therefore result in more accurate paratope prediction. The correlation of Paragraph's performance with ABlooper's model confidence also enables users to better understand when their predicted paratope is likely to be accurate.

Future work could expand and improve Paragraph by training on antibody models from more accurate structure prediction tools that have become available since Paragraph's original publication (Evans et al. 2022; Abanades et al. 2023; Lee et al. 2023; Ruffolo et al. 2023) or on multiple models from different tools to improve its robustness. These advancements in structural modelling could also allow variations of Paragraph to be trained to predict the binding sites of other immune proteins, such as nanobodies and TCRs.

Finally, Paragraph could be expanded to output multi-class predictions that describe the probabilities of residues forming different types of inter-atomic interactions, similar to proABC-2 (Ambrosetti et al. 2020a). Knowledge of the specific interaction types likely to be formed by each residue would be of particular use in computational docking experiments. However, training multi-class models may be hindered by even greater class imbalances than we faced and the same lack of antibody-antigen crystal complexes.

In the next chapter, we explore a well-studied antibody-antigen interaction (Trastuzumab-HER2) where the paratope is already known. In this work, we focus on optimising Trastuzumab's CDRH3 (known to contain four paratope residues) to maximise its binding affinity to HER2.

3 | Computational design and iterative improvement of diverse, high-affinity antibody libraries

3.1 Abstract

This chapter applies different computational tools to classify and design high-affinity variants of Trastuzumab, a cancer-fighting antibody therapeutic. We focus on ensuring these tools perform well when data is limited, suggesting they may be used for continuous, iterative library refinement in many research settings.

The following text is adapted from *Baselining the Buzz. Trastuzumab-HER2 Affinity, and Beyond!* (Chinery et al. 2024a). I prepared the training data, investigated the library design methods, and evaluated the FLAML and CNN affinity classification approaches. Alissa Hummer performed the EGNN analysis. Brij Bhushan Mehta led the collection of our new dataset, HER2-aff-large. Rahmad Akbar, Puneet Rawat, Andrei Slabodkin, Khang Le Quy, and Fridtjof Lund-Johansen supported the data collection and pre-processing. I wrote all sections excluding Section 3.3.4.3 (written by Alissa Hummer) and the Appendix's HER2-aff-large collection methods (written by Brij Bhushan Mehta).

3.2 Introduction

As described in Chapter 1, antibodies are critical proteins of the adaptive immune system and a powerful class of therapeutic. They are used to treat a variety of diseases including viral infections and cancer and their importance as therapeutics has steadily increased over recent years (Urquhart 2019; Lu et al. 2020). Developing antibody drugs however is expensive and time-consuming, often requiring over $1bn in investment and years of research and development, with no guarantee of success. Lead optimisation - where an initial drug candidate is engineered to improve its safety, efficacy, and developability - can dominate the pre-clinical stages of this pipeline (Paul et al. 2010).

Increasing the affinity of an antibody candidate for its target (antigen) is often a key aim of lead optimisation (see Section 1.2.4.2). Traditionally, Deep Mutational Scanning (DMS) has been favoured for this step (Whitehead et al. 2012). DMS involves introducing point mutations within, or near, the binding site (paratope) of an antibody and measuring their impact on binding. Beneficial mutations are then weighted more highly when designing antibody libraries for further testing.

DMS has proved effective in designing libraries highly enriched in binding variants (Mason et al. 2021). However, DMS adds additional time and costs to experiments, and can prematurely limit the subsequent sequence space explored. Recent developments in Machine Learning (ML) have sought to address some of these issues. These developments include 'one-shot' library design, where novel sequences are designed based on the knowledge of a single known binder.

One-shot library design

Hie *et al.* (Hie et al. 2023) trained a Large Language Model (LLM) (see Section 1.3.4.4) to suggest 'plausible' single-point mutations to a known binder, 14-71% of which were shown to improve affinity, depending on the starting 'wildtype' antibody and antigen. To obtain multi-point mutations, the top predicted single-point mutations first required experimental validation, similar to DMS. In this hybrid scenario, fewer single-point mutants needed testing, but the sequence space explored was limited.

Shanehsazzadeh *et al.* (Shanehsazzadeh et al. 2023) computationally suggested immediate multi-point mutations for Trastuzumab binding Human Epidermal Growth Factor Receptor 2 (HER2), a protein overexpressed in certain breast cancers. This work focused on replacing the third Complementarity Determining Region of Trastuzumab's heavy chain (CDRH3) with novel designs using both structure-design and inverse-folding models. This complete CDRH3 infilling approach produced an estimated 10.6% binder-enriched library and resulted in some sequences unlikely to be made following standard DMS. However, the majority of suggested mutations either simply restored germline residues, or were Glycine and Tyrosine substitutions.

LLMs have also been used to optimise other aspects of antibody design. Shuai *et al.* (Shuai et al. 2021) used their Immunoglobulin Language Model (IgLM) to optimise antibody humanness and developability. IgLM was built upon Open AI's Generative Pre-trained Transformer 2 (GPT-2) architecture (Solaiman et al. 2019) and was used to infill entire CDR loops, similar to Shanehsazzadeh *et al.* In offering immediate multi-point mutations, IgLM could potentially explore sequence space that DMS would not. However, like Shanehsazzadeh *et al.*, IgLM risks regressing to the germline due to the nature of the data it was trained on (Kovaltsuk et al. 2018; Olsen et al. 2022b; Olsen et al. 2024). Furthermore, this method has not yet been experimentally validated.

The ML methods listed above focus on designing antibody libraries from a single initial lead. These ML-designed libraries should contain more high-affinity (or more human) antibody variants than random mutations would offer, while also exploring different areas of sequence space than DMS-guided designs. However, designed libraries can be optimised further once some designs are tested experimentally. In this second step of the design process, ML classifiers can be trained on experimentally confirmed binders/non-binders and then used to design or pre-screen subsequent designs saving time and costs.

Large data affinity classification and library design

Li *et al.* (Li et al. 2023) used a pre-trained LLM, Gaussian Processes (GPs), and ensemble regression models to design and screen new high-affinity single-chain fragment variable antibodies (scFvs) against a conserved coronavirus peptide. They trained their models on experimental

data (26.5k heavy and 26.2k light chain sequences) that involved up to three random CDR mutations from an initial candidate. Using this approach, Li *et al.* computationally explored a relatively large portion of sequence space. However, the large amount of data used for training is often not practical.

Mason *et al.* (Mason et al. 2021) also used a large dataset, obtained by first performing a DMS screen of Trastuzumab CDRH3 mutants tested for binding to HER2. Using the DMS results, they generated a library and experimentally classified 36.4k mutants. They then trained a Convolutional Neural Network (CNN) classifier (see Section 1.3.4.2) using this data. The CNN was able to relatively accurately predict binding, achieving areas under the receiver operating characteristic and precision-recall curves (ROC AUC & PR AUC) of 0.91 and 0.83 respectively. However, like Li *et al.*, the amount of data used for training the CNN is not practical for many applications.

Finally, Akbar *et al.* (Akbar et al. 2022) and Frey *et al.* (Frey et al. 2023) used RNN-LSTMs (see Section 1.3.4.3) and Discrete Walk-Jump Sampling (a learned energy landscape explored using Markov-Chain Monte Carlo sampling) respectively to generate novel Trastuzumab variants against HER2. Each generative model was trained on ∼9k experimentally confirmed binding CDRH3 mutants (Mason et al. 2021). Akbar *et al.* achieved a predicted enrichment of 68%, while wet lab validation confirmed 70% of Frey *et al.*'s designs expressed and maintained affinity to HER2. These enrichments are higher than those reached by the general-purpose one-shot library design methods described earlier but they also require Mason *et al.*'s large labelled dataset for training.

In this chapter, we expand upon these previous developments in library design and affinity prediction, using Trastuzumab-HER2 as a model system. First, we describe a new, large dataset of over 524k mutations to Trastuzumab's CDRH3 for training and evaluating affinity classification tasks. We compare this new dataset to that collected by Mason *et al.*, examining how many variants are labelled as both binding and non-binding in each dataset. We then test different ML architectures for binding affinity classification, benchmark these methods, and test their accuracy when trained on small amounts of data. Finally, we use a variety of

computational methods to design novel, diverse antibody CDRH3 libraries against HER2 and computationally screen these using our highest accuracy classifier. Experimental validation of 700 of our top designed sequences using Biolayer Interferometry (BLI) is ongoing.

Together, our developments represent both a baseline for comparison of novel computational advances in this area and a lightweight end-to-end pipeline that can be used to speed up, enrich, and diversify antibody library design in many research settings (see Figure 3.2.1).

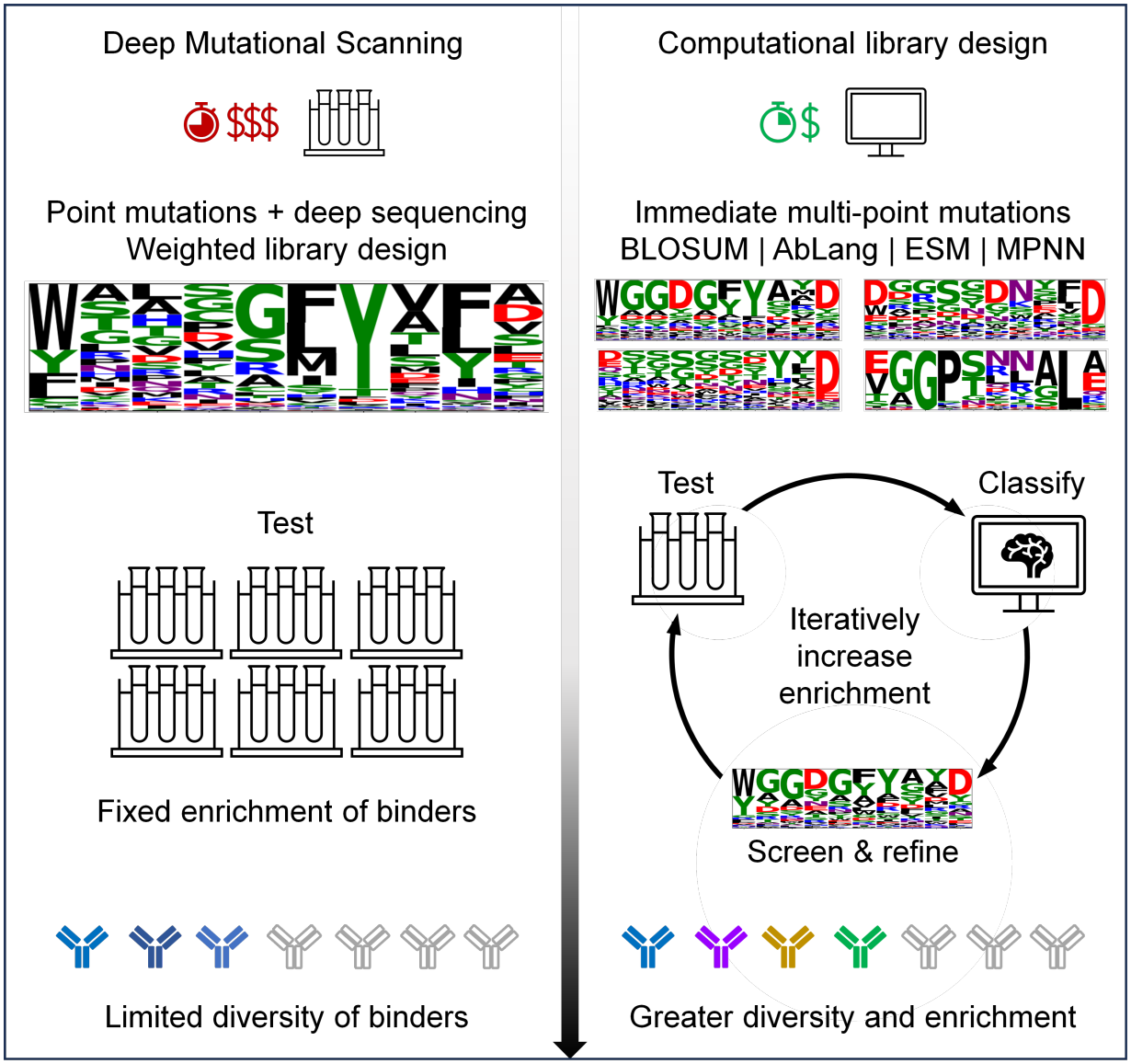

Figure 3.2.1: Overview of classical (left) and our proposed computationally-assisted (right) pipelines for discovering high-affinity variants of an initial antibody lead. The classical pipeline uses the results from Deep Mutational Scanning (DMS) to design an antibody library against a specific antigen. Thousands of sequences from this library are then experimentally tested, and enrichments of binding antibodies are usually found. Our proposed pipeline uses computational tools such as BLOSUM (Henikoff et al. 1992), AbLang (Olsen et al. 2022a), ESM (Rives et al. 2021), and ProteinMPNN (Dauparas et al. 2022), to design initial antibody libraries, without the need for DMS. These initial libraries are already enriched in binding antibodies above a random baseline, but this enrichment is increased further using continuous/active machine learning once experimental data starts to be collected. Our results demonstrate that machine learning classifiers can be trained on just a few hundred sequences to predict which antibodies are likely to bind. These classifiers are then used to screen untested sequences. This iterative process of experimental validation, retraining of the classifier, and refined screening of the antibody library results in highly enriched final libraries and less time and cost spent testing non-binding antibodies compared to standard experimental approaches. Furthermore, computational library design methods will produce sequences that DMS is unlikely to explore, resulting in a more diverse pool of binders.

3.3 Materials and Methods

3.3.1 HER2 affinity data

Our affinity classification methods were trained and evaluated on data for two distinct targets. The first and most varied of these datasets involves Trastuzumab variants tested for binding against HER2, a protein overexpressed in certain breast cancers.

The HER2 binding data used in this paper was collected from three sources - Mason *et al.*, Shanehsazzadeh *et al.*, and our own dataset - HER2-aff-large (see Results). We focused most of our analysis on HER2-aff-large as this dataset is larger than the others. To allow direct comparisons against previous analyses (Mason et al. 2021), we also show results on Mason *et al.*

3.3.1.1 Mason *et al.*

The Mason *et al.* dataset contains 38,733 mutated Trastuzumab sequences classified as positive (11,277) and negative (27,456) HER2 binders. Ten residues between IMGT (Lefranc et al. 2003) positions 107 and 116 (wildtype - `WGGDGFYAMD`) were mutated in their analysis, while all other positions were fixed to match Trastuzumab (we follow this approach in designing our dataset, see Results). Binding sequences were classified using mammalian display and Fluorescence Activated Cell Sorting (FACS, see Appendix). The ∼30% enrichment of binding sequences observed in their experiments is a result of single-site deep mutational scanning (DMS) being used to guide the library design.

Overlap is observed between the positive and negative classes (see Results). Mason *et al.* assign all overlapping sequences to the positive class in their analysis, resulting in 36,391 non-redundant sequences with a class imbalance of 31.0%. Choosing instead to remove any overlapping sequences completely results in a dataset of 34,049 sequences with a class imbalance of 26.2%. Throughout the chapter, we present our results using the latter approach, given the ambiguity of the overlapping sequences.

3.3.1.2 Shanehsazzadeh *et al.*

Shanehsazzadeh *et al.* recently published a dataset of 421 HER2 binding antibody sequences, validated using Surface Plasmon Resonance (SPR) (Shanehsazzadeh et al. 2023). No negative sequences were published. This dataset could therefore not be used for training and was only used as a test set.

Shanehsazzadeh *et al.* allow mutations between IMGT positions 105 and 117 (wildtype - `SRWGGDGFYAMDY`). Their designs also frequently delete one residue from Trastuzumab's CDRH3 and occasionally insert up to two additional residues. We limited our analysis to only those sequences that matched Trastuzumab in length (198 sequences). We did not enforce the matching of IMGT positions 105, 106, and 117 to Trastuzumab as this reduced the usable dataset size to only four sequences.

3.3.2 Influenza affinity data

In addition to anti-HER2 data, we test our methods on an anti-influenza dataset (see Appendix). The influenza data we used was obtained from Phillips *et al.* (Phillips et al. 2021) This dataset consists of variants of two broadly-neutralising anti-influenza antibodies (CR9114 and CR6261), with binding affinities measured against four subtypes of the influenza surface protein hemagglutinin (HA). This is one of the largest open-source datasets available and includes mutations made outside the CDR loops in the antibody heavy chain. However, it is still limited in scope with mutations made between the wildtype and germline amino acids only.

The CR9114 sub-dataset contains 65,536 variants (16 positions with all wildtype-germline mutation combinations). The CR6261 sub-dataset contains 2,048 variants (11 wildtype-germline combinations). The imbalance of binding vs. non-binding variants for each sub-dataset varied greatly, from 0.3% to 97%, when their affinity was measured against each of the four HA subtypes (see Appendix).

We trained our CNN affinity classifier on all available CR9114 antibody-HA-subtype combinations and compared our results to those from Bachas *et al.* (Bachas et al. 2022) (see Appendix).

3.3.3 Train-test dataset creation

For all HER2 tests, a train-validation dataset size ratio of 70-15 was used as in Mason *et al.* For each train-validation dataset, all remaining data was assigned to the test set. For each train-validation dataset size, we split the data both randomly and by clonotype. When splitting by clonotype, sequences were clustered according to their V and J genes, as annotated by ANARCI (Dunbar et al. 2016a), and by 70% sequence identity across the CDRH3. All HER2-aff-large sequences share the same V-gene (IGHV3-66) and one of two J-genes (IGHJ4 or IGHJ1). All members of a clonotype were added to the same train, validation, or test set. In all cases, we ensured train, validation, and test sets have the same class imbalances (i.e. ratio of binders to non-binders). When comparing classification models, such as FLAML and CNN (see Sections 3.3.4.1 & 3.3.4.2), we used identical train, validation, and test sets. All datasets can be found at doi.org/10.5281/zenodo.10549114.

3.3.4 Affinity classification methods

Three methods were trialled for classifying Trastuzumab variants based on their affinity to HER2 - FLAML (a decision-tree method), a CNN, and an EGNN. We aimed to use only existing, established architectures with default settings and identify those that performed well when trained on little data. Our three selected methods are described below and summarised in Figure 3.3.1.

3.3.4.1 FLAML Auto-ML

FLAML - a Fast Library for Automated Machine Learning (Wang et al. 2020) was used to provide a simple ML baseline. FLAML's default settings were used, allowing it to test multiple tree-based architectures, such as LightGBM, XGBoost, and random forest. Hyperparameter optimisation was also performed automatically.

For HER2 binders, FLAML took as input a one-hot encoding of the 10 CDRH3 residues between IMGT positions 107 and 116. When the mutated residues were dispersed throughout the antibody, as was the case for the Influenza data, a one-hot encoding of the entire sequence was used. FLAML requires 1D, not 2D, input; therefore, the one-hot encodings were flattened by

concatenating the individual residue encodings e.g. $20 \times 10 \rightarrow 200 \times 1$.

FLAML was allowed up to six hours to train and the best model was selected for testing in each instance. XGBClassifier performed best for all HER2-aff-large train and validation set sizes except the smallest (85) and largest (445,694) when the LGBMClassifier was selected. Early stopping was enabled. For training dataset sizes up to 3,000 sequences, less than five minutes was required to complete training on a CPU.

3.3.4.2 Convolutional Neural Network

We used a Convolutional Neural Network (CNN), adapted from Mason *et al.*, to classify binding and non-binding antibodies. The CNN used 400 convolutional filters, each of which had a kernel size of five and a stride of one. Similar to FLAML, for HER2 binders the CNN took as input a one-hot encoding of the 10 CDRH3 residues between IMGT positions 107 and 116. When the mutated residues were dispersed throughout the antibody, as was the case for the Influenza data, a one-hot encoding of the entire sequence was used. The CNN input was left as 2D (20×10 for HER2 binders) and padded with zeroes to ensure the filter outputs were the same length as the input. ReLU activations were used throughout.

To reduce overfitting, a dropout layer followed each convolutional filter. This layer set each convolutional output unit to zero with a probability of 0.2. The outputs of these dropout layers were pooled using max pooling with a pool size of two and a stride of one. No padding was used here, meaning the length of each output was one less than the input. These outputs were then flattened.

Finally, two dense fully-connected layers were applied which first reduced the flattened output to a vector of size 300 and then to size one. The first dense layer used ReLU activation, while the final used a Sigmoid activation function to limit the output to between zero and one. This output was the predicted probability that the sequence bound ($P = 1$) or did not bind ($P = 0$) a given target.

We used Adam optimiser, binary cross-entropy loss, a learning rate of 7.5×10^{-5}, and a batch size of 32 to train our network. The network was trained for a maximum of 100 epochs and

training was stopped if no decrease in loss was observed for five successive epochs. Training times ranged from less than a minute when trained on fewer than 3,000 sequences, and up to three hours when our entire HER2-aff-large dataset was used.

The CNN and training code were implemented in Python (v3.9) using TensorFlow (v2.15) and Keras (v2.15). Details of all dependencies used can be found at github.com/oxpig/Tz_her2_affinity_and_beyond.

Mason *et al.* report they performed hyper-parameter optimisation for the CNN architecture and training process. We performed no hyper-parameter optimisation beyond this.

3.3.4.3 Equivariant Graph Neural Network

In order to test the ability of a more complex method that takes into account protein structure, we applied an Equivariant Graph Neural Network (EGNN) architecture adapted from Hummer *et al.* (Hummer et al. 2023). The model takes as input a 3D structure of the Trastuzumab-HER2 complex and generates residue-level graphs. The graphs include the C_α atoms of the 10 mutated CDRH3 residues and surrounding neighbourhood (antibody C_α atoms within 10 Å of CDRH3 C_α atoms (antibody neighbourhood), antigen C_α atoms within 10 Å of the antibody neighbourhood and antigen C_α atoms within 10 Å of these antigen atoms). The node features are a one-hot encoded vector describing the residue type and chain type (antibody or antigen). The edge features are a one-hot encoded vector describing whether the edge is intra-binding partner, i.e. between atoms on the same binding partner, or inter-binding partner, i.e. between atoms on different binding partners.

The graphs are fed through a network composed of three EGNN layers (Satorras et al. 2021) with a hidden dimension of 128. The models were trained with Binary Cross-Entropy with Logits loss. The architecture was implemented using PyTorch and PyTorch Geometric.

To generate the structural inputs for the EGNN, we used FoldX BuildModel (Schymkowitz et al. 2005) to introduce mutations to the Trastuzumab CDRH3, starting from a FoldX-'repaired' structure in complex with HER2 (PDB 1N8Z (Cho et al. 2003)). As FoldX does not model changes to the backbone (Van Durme et al. 2011), the true structural effects of the mutations are

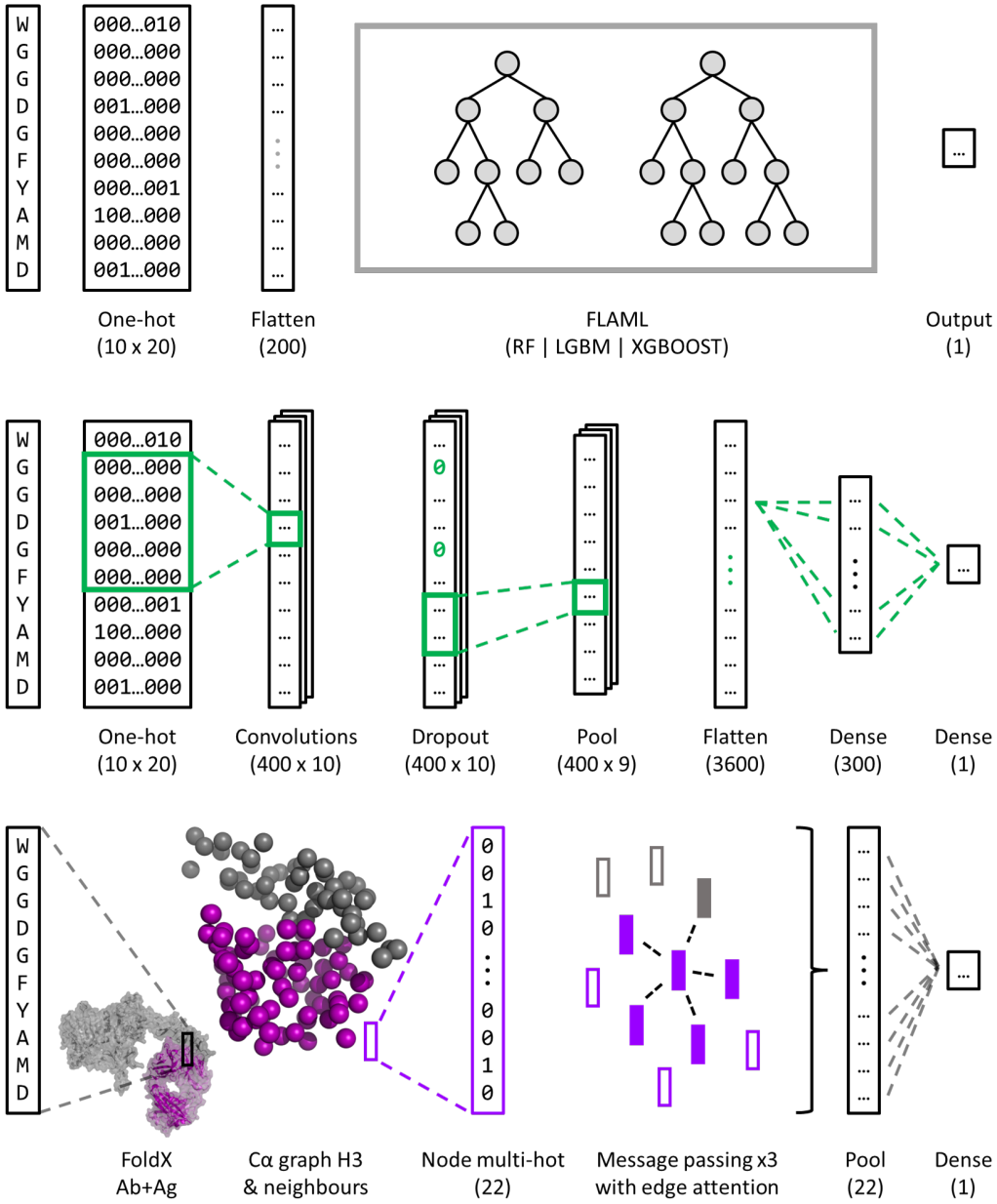

Figure 3.3.1: Overview of FLAML (top), CNN (middle), and EGNN (bottom) architectures. All models output a number between zero and one - the predicted probability that the input sequence binds the target in question. **FLAML** takes as input a flattened one-hot encoding of the mutated antibody sequence. FLAML trials multiple tree-based architectures and automatically selects the optimum one. The **CNN** takes as input a 2D one-hot encoding of the mutated sequence and passes this through a series of convolutional, dropout, pooling, and dense layers. The shape of the data at each step is shown in brackets. The **EGNN** uses a graph representation of the antibody (Ab, purple) and antigen (Ag, grey) in complex, focused on the CDRH3 neighbourhood. Nodes are located at the residues' C_α positions and their features describe the residue and chain type (Ab or Ag). Three message-passing layers are used with edge attention enabled. The final graph is pooled and classified using a series of dense layers.

unlikely to be represented. However, this approach has the advantages of speed (and therefore compatibility with high-throughput datasets) and avoiding the need for docking by starting from the structure of a bound complex.

The EGNN was trained for 100 epochs. GPU training times ranged from less than 3.6 hours, on datasets composed of fewer than 3,000 sequences, to ca. 20 days on the full HER2-aff-large dataset.

3.3.5 Computational library design methods

We used four open-source methods - BLOSUM (Henikoff et al. 1992), AbLang (Olsen et al. 2022a), ESM-2 (Rives et al. 2021), and ProteinMPNN (Dauparas et al. 2022) - to one-shot design antibody libraries against HER2, using Trastuzumab as an initial lead. For each method, we generated 1,000,000 sequences and aimed to retain 1,000 sequences that matched the edit distance distribution of HER2-aff-large (see Results and Appendix) i.e. the number of mutations each variant is from Trastuzumab. This sub-sampling allowed a fairer comparison between methods as smaller edit distances from Trastuzumab contain proportionally more high-affinity variants than large edit distances.

Some methods tended to generate sequences with larger edit distances from Trastuzumab due to the nature of their sampling distributions (see Figure 3.3.2 and Appendix). In these instances, the true number of sequences sampled at shorter edit distances was sometimes below the target number and this is stated where appropriate in our Results.

3.3.5.1 Random

As a baseline, we randomly mutated each of the ten Trastuzumab residues between positions 107 and 116 to each of the 20 standard amino acids with equal probability. The distribution of amino acids to sample from for each of the ten sequence positions was the same (see Figure 3.3.2).

3.3.5.2 BLOSUM

BLOSUM matrices (BLOcks SUbstitution Matrices) provide information on which amino acid substitutions are most likely to be observed (Henikoff et al. 1992). These matrices can be used to obtain the frequencies with which we expect to observe each amino acid type replaced with any of the standard 20 amino acids (Eddy 2004).

For our BLOSUM library design, we used the BLOSUM-45 matrix and, as background, the amino acid frequencies observed in CRDH3s in SAbDab (the Structural Antibody Database) (Dunbar et al. 2014; Schneider et al. 2022) in our reverse calculation (see Appendix). These choices aimed to tailor the general protein BLOSUM matrices for antibody use (see Appendix).

Once our final BLOSUM frequencies had been obtained, we used these to weight the sampling of each amino acid type based on the original Trastuzumab residue between positions IMGT 107 and 116. The distribution of amino acids sampled from was identical for matching amino acid types in Trastuzumab e.g. the Glycine residues at positions 108, 109, and 111 (see Figure 3.3.2).

3.3.5.3 AbLang

AbLang is an antibody language model designed to restore missing residues (Olsen et al. 2022a). AbLang was trained on over 14m sequences from the Observed Antibody Space database, OAS (Kovaltsuk et al. 2018; Olsen et al. 2022b). This set was dominated by germline sequences, so the restored residues often reflect germline observations.

We used AbLang to obtain amino acid likelihoods at each sequence position, independent of the original Trastuzumab residues. We tested two methods - masking the entire ten residues between IMGT positions 107 and 116 at once and masking just one residue at a time. In both instances, we limited AbLang to predicting CDRH3s of the same length as Trastuzumab. When all ten residues were masked at once, the likelihoods for each position were fixed i.e. once one masked position was infilled, the likelihoods were not recalculated for the remaining nine masked positions. Similarly, when masking one residue at a time, the likelihoods were calculated only once based on masking each position from the wildtype Trastuzumab CDRH3.

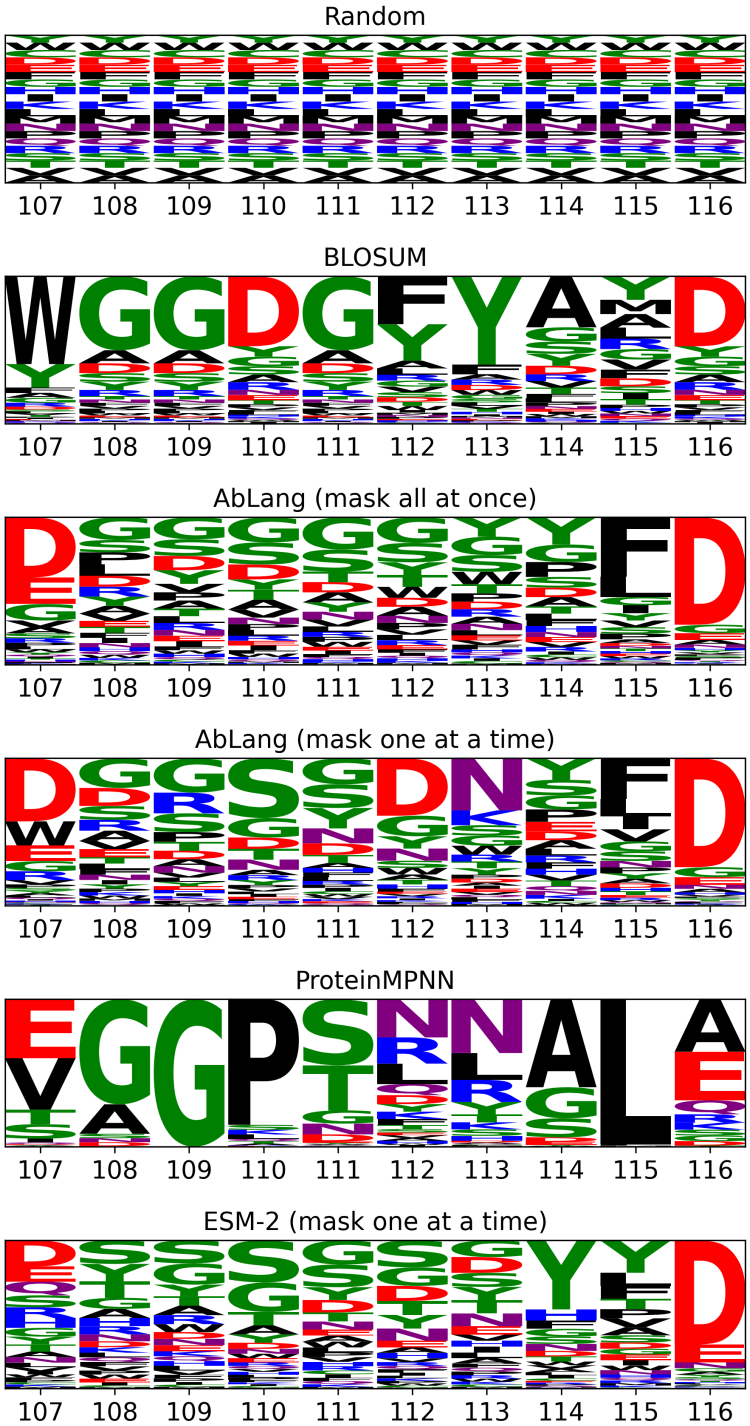

Figure 3.3.2: Logo plots of the raw weighted sampling distributions resulting from each method, specific to Trastuzumab (wildtype - `WGGDGFYAMD`). Each method is restricted to suggesting mutations between IMGT positions 107 and 116 (x-axis). The y-axis labels have been omitted for clarity, but runs between zero and one, on a linear scale. Residue codes that are 'tall' at a given sequence position have high likelihoods of being selected when designing new CDRH3 loops with that method.

The likelihoods returned by AbLang were used to weight the sampling of each amino acid type between positions IMGT 107 and 116. AbLang's likelihoods are position-specific, meaning the sampling weights were unique for each sequence position, unlike BLOSUM (see Figure 3.3.2).

3.3.5.4 ESM-2

Evolutionary Scale Modelling (ESM) is a general-purpose protein language model (not fine-tuned for studying antibodies) from Meta's Fundamental AI Research (FAIR) Protein Team (Rives et al. 2021). Recent methods have found success in using ESM to suggest affinity-improving single-point mutations (Hie et al. 2023); here we test its efficacy for multi-site mutations.

Like AbLang, ESM-2 can be used to obtain amino acid likelihoods at masked sequence positions. We masked each residue between IMGT positions 107 and 116 one at a time and used the 33-layer, 650m parameter implementation of ESM-2 (esm2_t33_650M_UR50D) to suggest residue likelihoods. We also tested masking the entire ten residues (IMGT positions 107 to 116) at once, limiting ESM-2 to predicting CDRH3s of the same length as Trastuzumab. However, predicted enrichments when masking all residues at once were lower than when masking residues one at a time (see Appendix), so we focus only on the latter in this chapter.

The likelihoods returned by ESM-2 were used to weight the sampling of each amino acid type between positions IMGT 107 and 116. ESM-2's likelihoods are position-specific, like AbLang (see Figure 3.3.2).

3.3.5.5 ProteinMPNN

ProteinMPNN (Dauparas et al. 2022) is an inverse-folding method for predicting protein sequences for a given structure. We used the ABodyBuilder2 (Abanades et al. 2023) predicted structure of Trastuzumab as the base structure as any structure-based method must work with a predicted structure as crystal structures are not readily available for most antibodies. CDRH3 residues between IMGT positions 107 and 116 were then masked, and ProteinMPNN was tasked with generating sequences predicted to fit the modelled CDRH3 conformation.

We used a high sampling temperature of 0.3 for ProteinMPNN to produce diverse sequences.

Unlike the previous methods described which were used to generate likelihoods to sample from, we used the exact sequences predicted by ProteinMPNN for our generated library (see Figure 3.3.2). Due to speed limitations and an observed lack of diversity, we generated only 3,000 sequences with ProteinMPNN, which resulted in 2,331 non-redundant outputs.

Considering the constrained distribution of sequences created by ProteinMPNN, it was not possible to sub-sample from these to match the edit distance distribution of HER2-aff-large. Instead, 1k sequences were randomly sampled from the 2,331 designed by ProteinMPNN for comparison against other methods. Further details can be found in the Appendix.

3.4 Results

3.4.1 HER2-aff-large offers a large, clean affinity-labelled dataset of Trastuzumab variants

We use Trastuzumab-HER2 as a model system to explore improvements to computational antibody library design and iterative binder enrichment. We also evaluate our methods on a smaller dataset of broadly neutralising anti-influenza antibodies (Phillips et al. 2021), results of those tests are given in the Appendix.

Our HER2-aff-large dataset contains over half a million Trastuzumab variants split into 'high', 'medium', and 'low' classes based on their binding affinity to HER2. As in Mason *et al.*, a previous study of Trastuzumab-HER2 binding affinity (see Methods), mutations were limited to IMGT positions between 107 and 116 of the heavy chain only. HER2-aff-large's library design was built upon the DMS results from Mason *et al.* (see Methods) and contains 178,160, 196,392, and 171,732 high, medium, and low-affinity binders, respectively (see Table 3.4.1).

A small percentage of overlap between affinity classes was observed in HER2-aff-large, with 1.1% of 'high' sequences also found in the 'medium' class, and 2.9% found in the 'low' class. This inter-class overlap is lower than that observed in Mason *et al.* (see Table 3.4.1). Assigning overlapping sequences to the 'high' class only, and removing any redundancy between 'medium' and 'low' classes results in a total dataset of size 530,357 and a class imbalance of 33.6%. Removing any overlapping sequences completely results in a dataset of 524,346 sequences with a class imbalance of 32.8%. We use this latter approach throughout this chapter.

In our analyses, we observe that the 'medium' and 'low' classes of HER2-aff-large cluster with the negative binders from Mason *et al.* using tSNE visualisations (see Figure 3.4.1). Furthermore, we observe that classification methods trained on data from Mason *et al.* offer low predicted binding probabilities for both 'medium' and 'low' classes, and high probabilities for the 'high' class (see Figure 3.4.4). Due to these observations, we assign 'high' affinity sequences to be positive binders and group 'medium' and 'low' affinity sequences to be negative binders for

binary classification (see Table 3.4.1). This aligns broadly with the goal of selecting high-affinity antibodies during lead optimisation.

Mason *et al.*

label	# total	# unique	positive overlap	negative overlap
positive	11,300	11,277	-	**20.76%**
negative	27,539	27,456	8.53%	-

HER2-aff-lrg

label	# total	# unique	positive overlap	negative overlap
positive	178,160	178,160	-	3.37%
negative	368,124	362,549	1.66%	-

Inter

Mason *et al.* label	# total	# unique	HER2-aff-large positive overlap	HER2-aff-large negative overlap
positive	11,300	11,277	110	**77**
negative	27,539	27,456	**29**	46

Table 3.4.1: A breakdown of Mason *et al.* (top) and HER2-aff-large (middle) HER2 binding affinity datasets. Mason *et al.* split their dataset into two classes - positive and negative. Our HER2-aff-large dataset is split into three classes - high, medium, and low-affinity binders. In our analyses, we assign high-affinity binders to be our positive class and group the mid and low-binding affinity classes into a single negative class. This binary separation aligns with the goal of selecting high-affinity antibodies during lead optimisation. Some inter-class redundancy is expected within these datasets given the relatively low precision of the high throughput experimental methods used. More inter-set redundancy is observed in Mason *et al.*'s data than in HER2-aff-large. Little overlap exists between Mason *et al.*'s data and HER2-aff-large due to the large possible sequence space ($\sim 10^{13}$, 20 possible amino acids and 10 sequence positions). As this overlap is small, we show the absolute number of overlapping sequences here instead of percentages. We find 41% of 187 overlapping sequences labelled as binding in Mason *et al.*'s dataset are labelled negatively in our HER2-aff-large dataset. Overlaps with contrasting classes greater than 20% are shown in bold. We focus our analysis on the HER2-aff-large dataset due to its order-of-magnitude larger size and smaller inter-class overlap.

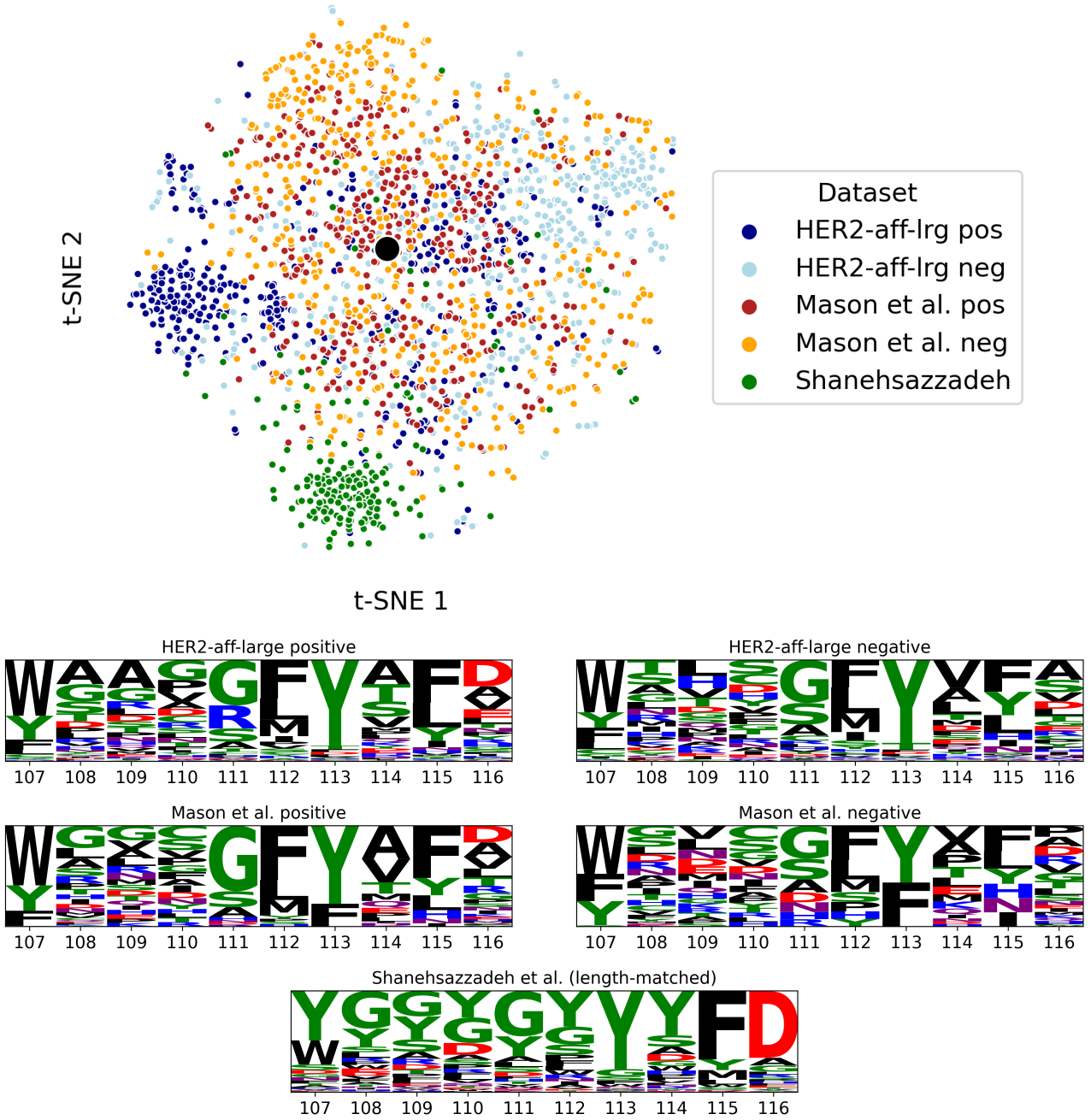

Figure 3.4.1: A comparison of all HER2-binding datasets. The top figure shows a tSNE visualisation of all of Shanehsazzadeh *et al.*'s 198 Trastuzumab-length-matched designs (all binding HER2) along with 500 sequences randomly sampled from the positive and negative members of Mason *et al.*'s dataset and HER2-aff-large. The tSNE uses one-hot encoded CDRH3 sequences as input. Trastuzumab is shown as a large black circle in the centre of the plot. The logo plots of all data from HER2-aff-large and Mason *et al.* are also shown alongside the logo plot of the same 198 sequences from Shanehsazzadeh *et al.* Visual inspection of all plots shows greater separation of HER2-aff-large positive and negative sequences compared to those from Mason *et al.* Meanwhile, Shanehsazzadeh *et al.*'s designs occupy a unique, narrow area of sequence space away from data from both HER2-aff-large and Mason *et al.*'s data.

3.4.2 CNN classifies high-affinity antibodies with little training data

We present our results in reverse order to our graphical abstract (see Figure 3.2.1) i.e. affinity classification before computational library design. We present our results in this order as our trained binding affinity classifier was used to computationally screen our newly designed libraries before experimental validation which is currently ongoing.

To be useful in most real-world settings, binder classification methods must perform well with limited data availability. This requirement is essential for continuous learning and iterative refinement of library design in subsequent rounds of experimentation.

We tested three classification methods - a Fast Library for Automated Machine Learning (FLAML) (Wang et al. 2020), a Convolutional Neural Network (CNN) (Mason et al. 2021), and an Equivariant Graph Neural Network (EGNN) (Satorras et al. 2021) - using varying amounts of data for training and validation. Full architecture details can be found in our Methods. Data was assigned to train, validation, and test sets randomly, but we ensured class imbalances remained consistent between each. Results are also presented with train, validation, and test sets split by clonotype (see Methods).

The CNN outperformed both FLAML and the EGNN when trained on small amounts of data, up to 2,000 sequences (see Figure 3.4.2a). The low training data requirement of the CNN (achieving a PR AUC of 0.71 when trained on only 170 sequences) offers the potential to iteratively increase experimental enrichments of binding antibodies through continuous learning as more data is collected. PR AUC is used as our main evaluation metric as it correlated with ROC AUC for all methods but offered a greater range of discriminatory values (see Appendix). Chapter 2 also showed PR AUC to be well suited for evaluating class imbalanced problems where the minority class (HER2-binding variants in this instance) is unevenly distributed e.g. in sequence space. All classifiers were optimised for ROC AUC or Accuracy to avoid potential training biases (McDermott et al. 2024). Further evaluation metrics and approximate training times can be found in the Methods and Appendix.

Beyond the low data regime, FLAML outperformed the CNN for intermediate train set sizes,

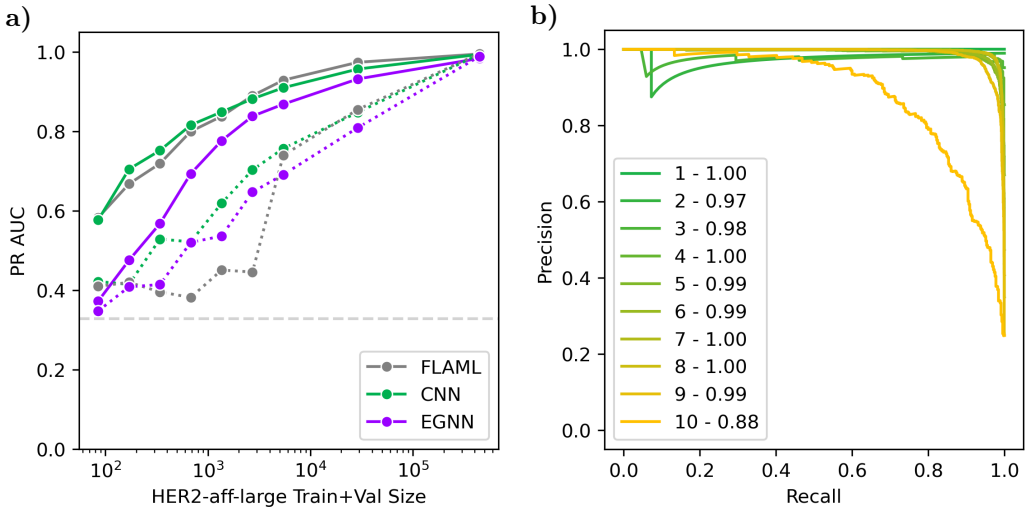

Figure 3.4.2: a): Areas under the Precision-Recall curves (PR AUC) for all binder classification methods (FLAML, CNN, and EGNN). Results are shown on our HER2-aff-large dataset with overlapping sequences removed. Train, validation, and test sets are split randomly (solid) and by clonotype (dashed). Train and validation sets have a relative size ratio of 70:15. All data not assigned to the train or validation dataset is used as the test set, on which the results are presented. Random guessing would result in PR AUC values equal to HER2-aff-large's class imbalance of 0.33 (light grey dashed line). **b)**: Precision-Recall curves for variants grouped by edit distance from Trastuzumab for the CNN trained on all HER2-aff-large data, using a 70-15-15 random split (CNN-HER2-max). Results are given on the corresponding test set. The legend shows the edit distances and corresponding PR AUC values. The dips in precision at recall ∼0.1 for edit distances two and three occur due to rare misclassifications within the small test sets of these edit distances.

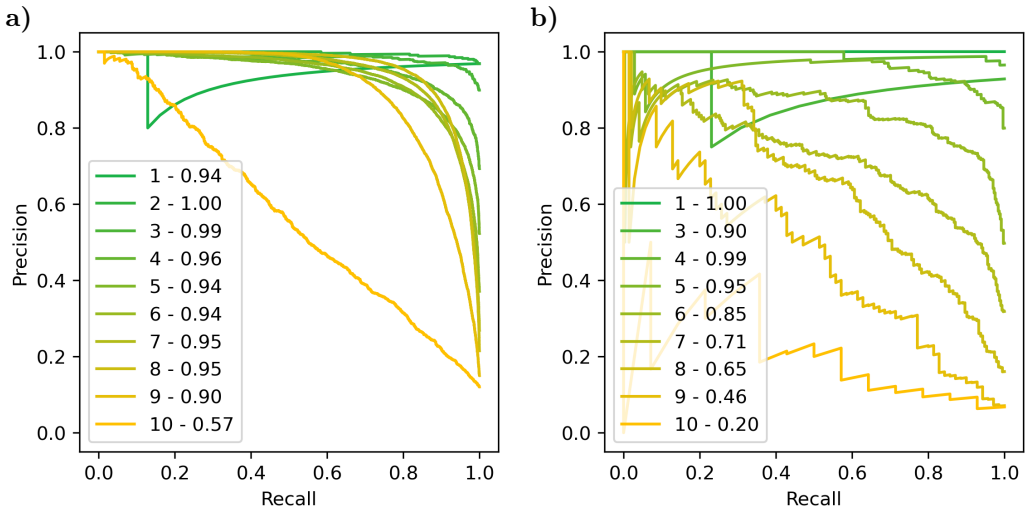

Figure 3.4.3: a): Precision-Recall curves for variants grouped by edit distance from Trastuzumab for the CNN trained and validated on 28,941 sequences from HER2-aff-large data, sub-sampled to match Mason *et al.*'s class imbalance of 26.2% Results are presented on the corresponding HER2-aff-large test set. **b)**: Precision-Recall curves for variants grouped by edit distance from Trastuzumab for the CNN trained and validated on 28,941 sequences from Mason *et al.*'s data. Results are presented on the corresponding Mason *et al.* test set. The legends show the edit distances and corresponding PR AUC values for both figures ('edit - PR AUC'). No sequences of edit distance two were present in Mason *et al.*'s data. The same architecture and hyperparameters were used for training on both sets of data, which Mason *et al.* optimised in their previous work.

but the performances converged when all available training data was used. Our subsequent analyses focus on the CNN due to its superior performance on small datasets, especially when the data is split by clonotype (see Figure 3.4.2a).

One obvious way for a predictor to perform well in this setting is for it to favour sequences with shorter edit distances (fewer mutations) from Trastuzumab. In the training data 98% of antibodies with edit distance of one from Trastuzumab were labelled as binding, while only 17% with an edit distance of ten bound. When the CNN was trained on all available HER2-aff-large data, using a 70-15-15 split, we observed near-perfect accuracy for edit distances one to nine (see Figure 3.4.2b). Antibodies with an edit distance of nine have a similar class imbalance (19%) to those with an edit distance of ten (17%), but the PR AUC is substantially higher (0.99 vs 0.88). This disparity indicates some knowledge of the original sequence is of benefit to the trained network.

3.4.3 Classification accuracy varies when trained on data from different experiments

We investigated the performance of the CNN architecture when trained on different data sources. We trained our CNN on the same amount of data from both HER2-aff-large and Mason *et al.* datasets and evaluated the performance on the corresponding test sets (see Figure 3.4.3). To allow a fair comparison, we randomly sub-sampled positive binding sequences from HER2-aff-large so that the class imbalances between the datasets were the same (26.2%). We also confirmed that the datasets contained similar edit distance distributions.

When trained (using 28.9k sequences) and tested on Mason *et al.*'s data, the CNN achieved a PR AUC of 0.84, compared to a PR AUC of 0.94 when trained and tested on HER2-aff-large (again, using 28.9k training sequences). Furthermore, the performance of the CNN fell sharply with increasing edit distance from Trastuzumab when trained and tested on data from Mason *et al.* This effect was less strong when the CNN was trained and tested on HER2-aff-large, despite this dataset containing proportionally fewer sequences with edit distances of nine and ten compared to the data from Mason *et al.* The lower accuracy of the CNN when trained and tested on data from Mason *et al.* may suggest the dataset contains higher levels of noise i.e.

sequences incorrectly labelled as belonging to either the positive or negative class.

3.4.4 Trained classifiers do not transfer well between experiments

To test the robustness of the CNN affinity classifier, we trained on data from HER2-aff-large and tested on data from Mason *et al.*, and vice versa. We also evaluated both trained CNNs on the 198 Trastuzumab-length-matched binding variants released by Shanehsazzadeh *et al.* (Shanehsazzadeh et al. 2023). We trained and validated the CNN on the same number of sequences from both datasets (28,941) to allow a direct comparison.

When tested on data from an experiment different from the training data, the predictive power dropped compared to evaluation on data from the same experiment (see Figure 3.4.4). This drop in performance was greater than the fall seen when splitting a dataset by clonotype (see Figure 3.4.2a), indicating that the drop was not caused by exploring different areas of sequence space alone. Instead, differences in experimental set-ups and/or cut-offs used to separate positive and negative classes may contribute to the poor generalisability.

As HER2-aff-large offered the largest dataset of HER2 binding data available and also produced

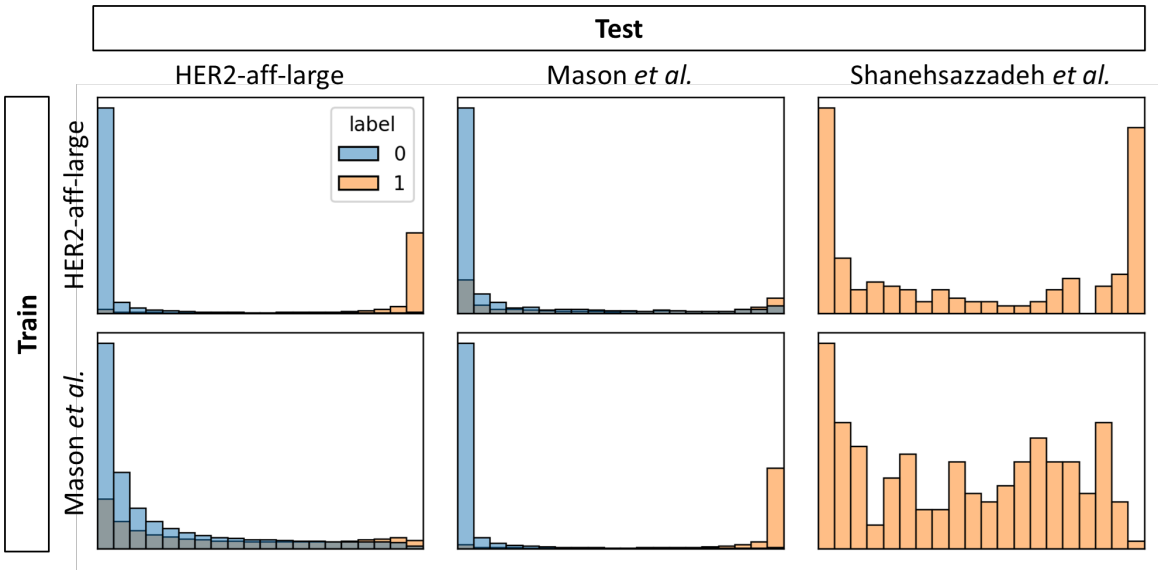

Figure 3.4.4: Histograms showing CNN predictions when trained and tested on different datasets. The x-axis runs from zero to one. The y-axis measures the number of sequences recorded in each prediction bin. Binding sequences are labelled 'one' and shown in orange. Non-binding sequences are labelled 'zero' and shown in blue. Axis labels have been omitted for simplicity. The accuracy of each trained CNN drops when tasked with classifying sequences from experiments outside the training data.

the best models (highest PR AUC), we used our CNN trained on all this data (CNN-HER2-max) to screen our novel computational library designs before experimental validation.

3.4.5 BLOSUM, AbLang, ESM, and ProteinMPNN generate antibody libraries with high proportions of predicted binders

We next explored the ability of computational methods to complement or replace DMS for library generation (see graphical abstract, Figure 3.2.1). Computational methods, if able to replace DMS, could design more diverse antibody libraries and save initial experimental time and money.

Using our CNN trained on all HER2-aff-large data (CNN-HER2-max), we predicted the binding probabilities of Trastuzumab CDRH3 variants generated using BLOSUM (Henikoff et al. 1992), AbLang (Olsen et al. 2022a), ESM (Rives et al. 2021), ProteinMPNN (Dauparas et al. 2022), and a random baseline. Briefly, BLOSUM matrices measure whether or not an amino acid substitution is conservative, AbLang and ESM are LLMs that output masked-residue likelihoods, and ProteinMPNN designs sequences for a given structure (see Methods for more details). These computational tools were used to design diverse, length-matched Trastuzumab CDRH3 libraries without relying on the results of any experiments, effectively zero-shot (AbLang - mask all at once) and one-shot (BLOSUM, ProteinMPNN, ESM-2, AbLang - mask one at a time) prediction of the CDRH3.

To allow fair comparisons to experimental data, for each library design method we aimed to sub-sample 1,000 sequences from a maximum of 1,000,000 generated sequences to match the edit distance distribution observed in HER2-aff-large (see Appendix). This sub-sampling is required as smaller edit distances contain proportionally more high-affinity variants than larger edit distances.

The exact libraries generated varied depending on the starting seed, but some methods consistently failed to explore small edit distances. This restriction meant the total counts of AbLang and ESM sub-sampled libraries were below 1,000 but above 985. Our Random library contained just 920 sequences. ProteinMPNN only sampled edit distances of five to nine, so no edit

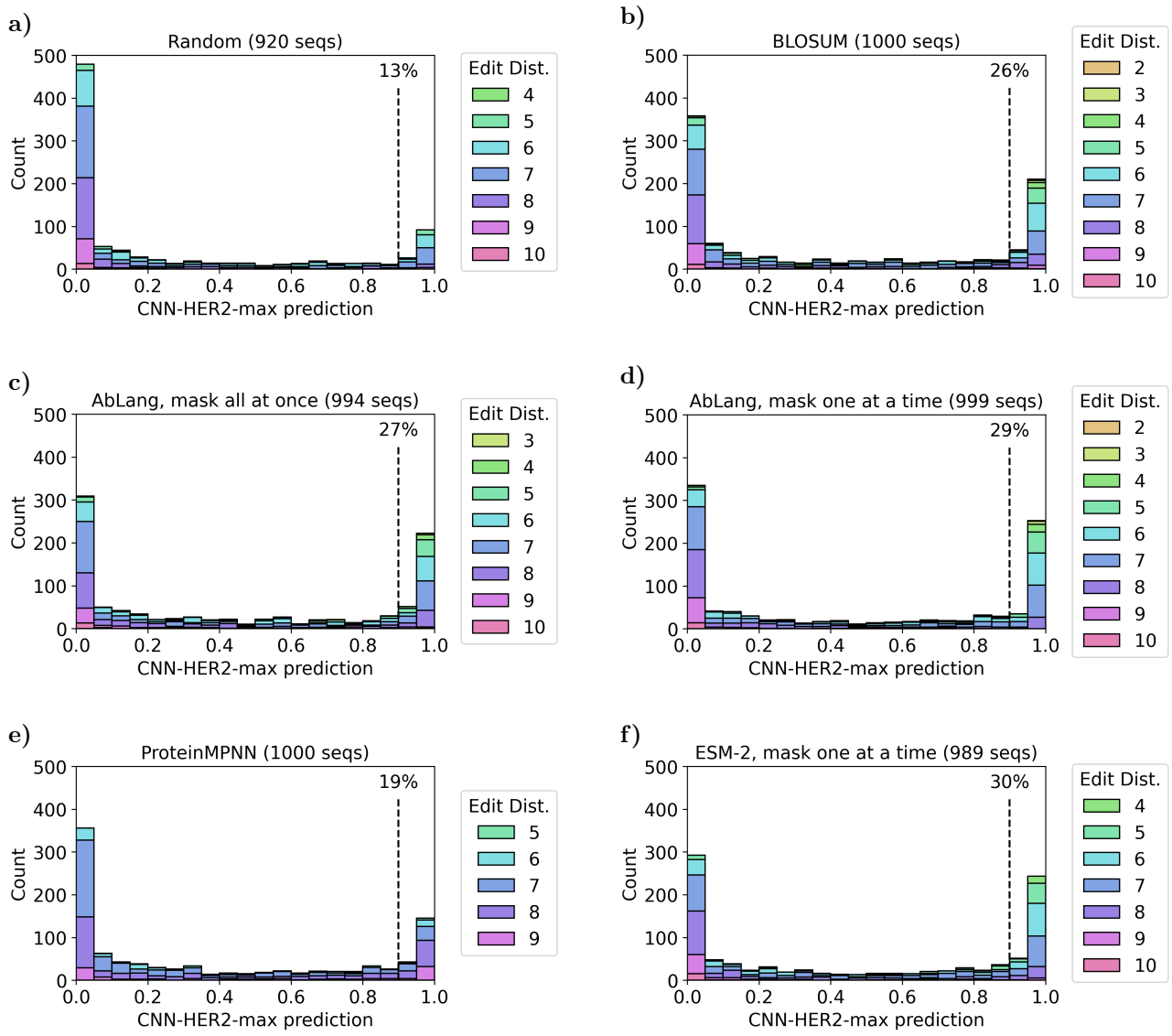

Figure 3.4.5: Distributions of HER2 binding predictions for ∼1k sequences generated using BLOSUM, AbLang, ESM, and ProteinMPNN, plus a random baseline. These sequences were sub-sampled from a maximum of 1m generated sequences to match the edit distance distribution observed in HER2-aff-large, where possible. Some methods failed to produce any sequences at certain edit distances - when an edit distance is not present in the legend, no sequences of this edit distance were generated in the 1m attempts, and the total count (shown in the figure titles) is slightly less than 1,000. Binding predictions (x-axis) are those given by our CNN trained on all HER2-aff-large data (CNN-HER2-max), using a 70-15-15 split. A value close to '1' indicates a high predicted binding probability, while a value close to '0' indicates a low probability of binding. Dashed lines are drawn at 90% CNN-HER2-max predicted binding probabilities and the percentage of sequences with binding probabilities above this cut-off are stated above the line. All methods succeeded in producing some sequences with high predicted binding probabilities to HER2 across a range of edit distances.

distance restrictions were included for this method. Full details of how these sequences were generated can be found in our Methods. Predicted enrichments when enforcing edit distance distributions to match those of ProteinMPNN can be found in the Appendix.

We found a large proportion of sequences generated using all methods to have high CNN-HER2-max predicted binding probabilities (see Figure 3.4.5). For Random, BLOSUM, AbLang (mask all ten residues at once), AbLang (mask one residue at a time), ProteinMPNN, and ESM (mask one residue at a time), we found 13%, 26%, 27%, 29%, 19%, and 30% of sequences to have CNN-HER2-max binding probabilities above 90%, respectively. To determine how many sequences actually bind, Biolayer Interferometry experiments are being run for 140 sequences designed by each method.

3.4.6 Predicted binder enrichments remain high when generating libraries from different starting sequences

One-shot library design methods, such as BLOSUM, ProteinMPNN, and AbLang/ESM (masking one residue at a time), require knowledge of an initial binding sequence. For most targets, an initial lead is often known or easily obtained through library screens. However, this lead may not have as high affinity as Trastuzumab-HER2.

To test the robustness of our library design methods to different starting sequences, we designed libraries using various randomly selected high-affinity sequences from HER2-aff-large as our starting point. These new libraries were then classified using CNN-HER2-max as before (see Table 3.4.2). Designs for AbLang (mask all at once) are independent of the starting sequence and so are excluded from this analysis.

For most new starting sequences, predicted enrichments fell for all methods compared to starting with Trastuzumab. Nevertheless, predicted enrichments often remained above that achieved by our random baseline (13%). ProteinMPNN's designed libraries showed the greatest variation of predicted enrichments compared to all other methods, while AbLang (mask one at a time) achieved the most consistently high performance (see Table 3.4.2).

Sequence	Edit	BLOSUM	AbLang	ESM	MPNN
WGGDGFYAMD	0	26%	29%	**30%**	19%
WGGDGFYANS	2	21%	**32%**	26%	14%
WGRFGFYALL	4	13%	25%	19%	**46%**
WDGARLYNLD	6	**27%**	28%	22%	33%
WGLAILFTTS	8	14%	18%	7%	15%
VLAGVRAEDT	10	12%	17%	13%	5%

Table 3.4.2: A comparison of predicted enrichments for libraries designed from various starting sequences using BLOSUM, AbLang (mask one residue at a time), ESM (mask one residue at a time), and ProteinMPNN. The table shows the starting CDRH3 sequences used in each instance and their edit distances from Trastuzumab. For each method, ∼1k sequences were sub-sampled from a maximum of 1m generated sequences to match the edit distance distribution observed in HER2-aff-large, where possible. The percentages show the proportion of these designs that had CNN-HER2-max binding probabilities above 90% (higher is better). The largest predicted enrichments for each method are shown in bold.

As before, for each library design method (excluding ProteinMPNN) we aimed to sub-sample 1k sequences from a maximum of 1m generated sequences to match the edit distance distribution observed in HER2-aff-large.

3.4.7 BLOSUM, AbLang, ESM, and ProteinMPNN's predicted binders cover diverse areas of sequence space

The sequences predicted to bind HER2, designed by all methods using Trastuzumab as a starting sequence, covered diverse areas of sequence space (see Figure 3.4.6), with many large edit distances away from Trastuzumab's original sequence (SRWGGDGFYAMDY, IMGT positions 105 to 117; see Figures 3.4.5 & 3.4.6c).

Many of the designed sequences were also large edit distances from the closest germline (AR----YFDY, IMGT positions 105 to 117; see Figure 3.4.6c), as given by the V and J genes (IGHV3-66 and IGHJ4) assigned using ANARCI (Dunbar et al. 2016a), with the corresponding sequence taken from imgt.org. The residue positions indicated by '-' are non-germline positions.

Finally, most designed sequences and confirmed binders occupied areas of sequence space far from our observed data (HER2-aff-large) as visualised using a tSNE plot (see Figure 3.4.6a,b).

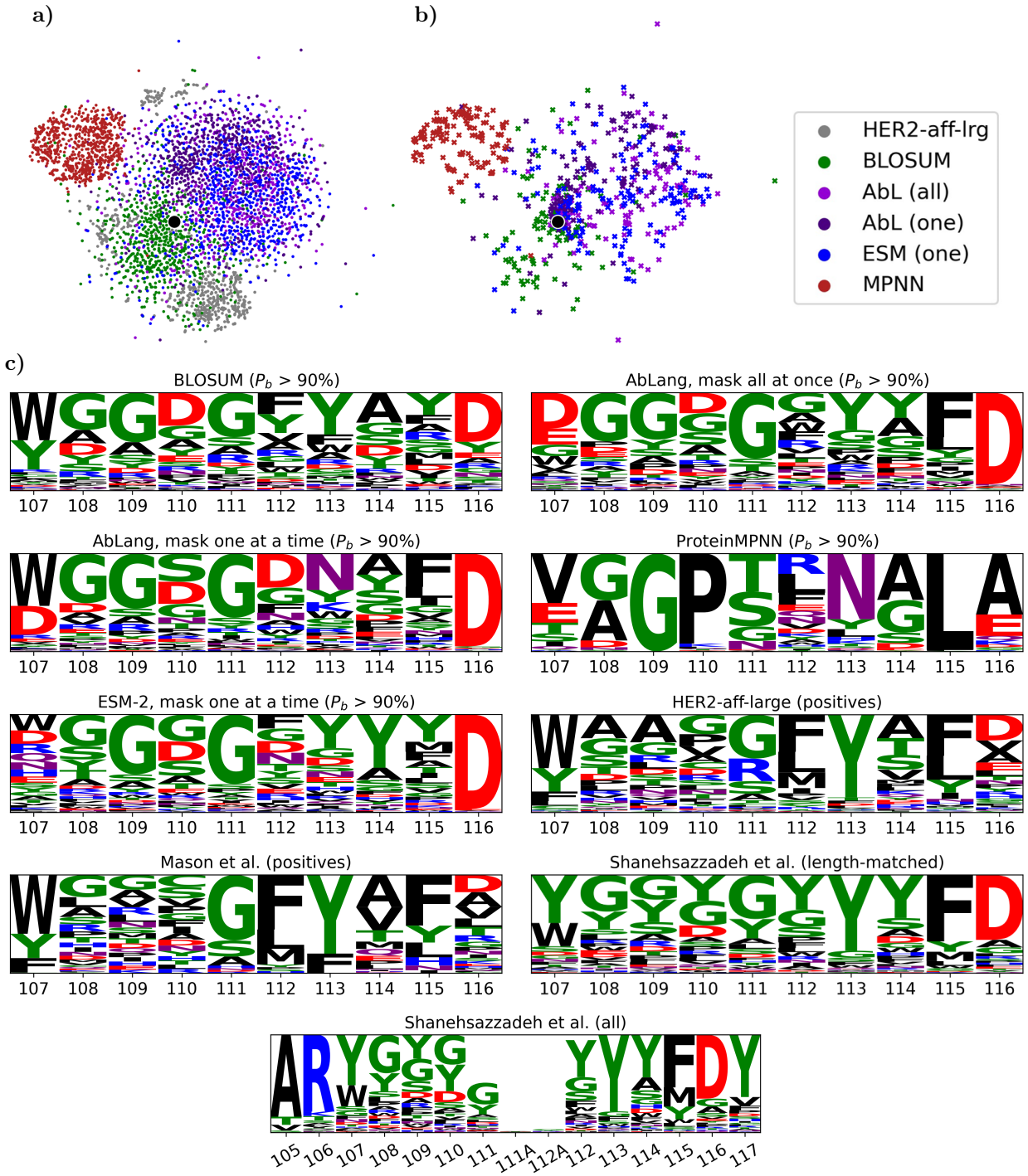

Figure 3.4.6: **a)** t-SNE plot comparing the sequence space explored by different methods when designing HER2 binders. ∼800 sequences designed by each method are shown and compared to HER2-aff-large. Our t-SNE takes as input the flattened one-hot encodings of the designed CDRH3 loops (residues 107 to 116). **b)** t-SNE of sequences shortlisted for experimental validation with CNN-HER2-max binding probabilities above 90%. Both t-SNEs use the same scale. Outlying data points are cropped from **a)** for plotting clarity. Trastuzumab is shown in black. **c)** Logo plots for BLOSUM, AbLang, ESM, and ProteinMPNN HER2 predicted binders. All positive experimental data are shown for comparison.

PCA and UMAP visualisations showing similar distributions can be found in the Appendix. These newly confirmed binding sequences were unlikely to have been designed following DMS experiments due to DMS weighting the search space based on observations from single-point mutations. HER2-aff-large's comparatively small search area in Figure 3.4.6a (grey) highlights the restrictive nature of DMS.

Despite their diversity, some shared features were observed between most computationally designed antibody libraries, including a tendency towards Glycines at IMGT positions 108, 109, and 111, and Aspartic acid at position 116. These shared features are observed in Trastuzumab's original CDRH3, and some are similar to those in HER2-aff-large and Mason *et al.*'s data (see Figure 3.4.6c), which suggests that they are beneficial for HER2 binding.

ProteinMPNN occupies a distinct area of the tSNE visualisation in Figure 3.4.6a, and its logo plot stands out from those of all other methods (see Figure 3.4.6c). This could be because ProteinMPNN is the only structure-based design method we include, or because it is a general-purpose protein sequence design method. However, ESM is also a general-purpose protein sequence design method and it designed sequences similar to those from AbLang (see Figure 3.4.6c).

3.4.8 Simulations show rapid increases in enrichment are achievable without sacrificing library diversity through active learning and continuous iterative library refinement

Figure 3.4.7 demonstrates how we can combine both computational library design methods, such as BLOSUM, AbLang, ESM, and ProteinMPNN, and a simple ML classifier, such as our CNN, to design and iteratively improve diverse antibody libraries, as originally suggested in Figure 3.2.1.

Our previous results show how computational zero-shot design methods, such as AbLang (mask all ten residues at once), can generate libraries with high predicted binder enrichments, ∼25%. These initial libraries are diverse (see Figure 3.4.6) and contain more binding variants than one would expect from random mutations, yet this enrichment can be increased further still.

Using a reduced form of HER2-aff-large with just 10% enrichment, we demonstrate how the iterative retraining of a CNN classifier can quickly increase the starting enrichment above that achieved by DMS (see Figure 3.4.7). We chose 10% starting enrichment assuming that not all predicted binders will bind ($< \mu$M K_D, in agreement with Shanehsazzadeh *et al.*). In similar work, Porebski *et al.* (Porebski et al. 2023) experimentally confirm that approximately half of their designs with $P_b > 90\%$ bound with greater Fluorescence Intensities (FI, shown to correlate with K_D) than a confirmed 320nM design. Should we observe similar results, this would correspond with true floor enrichments ∼10%. Biolayer Interferometry experiments (ongoing) will confirm our true success rate for each library design method.

Our simulations show that after testing just 540 sequences, subsequent rounds of experimentation contained libraries with enrichments above 30%. Enrichments continued to increase rapidly up to and above 80% after testing fewer than 2k sequences. This continuous retraining of the CNN and screening of new sequences did not limit the sequence search space but instead explored it in a more time and cost-efficient manner (see Figure 3.4.7). All areas of the tSNE plot containing positive data continued to receive hits in the latter rounds of enrichment. Additionally, the CNN did not learn to favour shorter edit distances from Trastuzumab, with the distribution of edit distances seen in latter rounds closely matching that of HER2-aff-large's positive data.

Crucially, this cycle of continuous learning and iterative library refinement is independent of the initial library design method. Such an approach could therefore allow faster and/or cheaper exploration of high-affinity antibody variants across diverse sequence space.

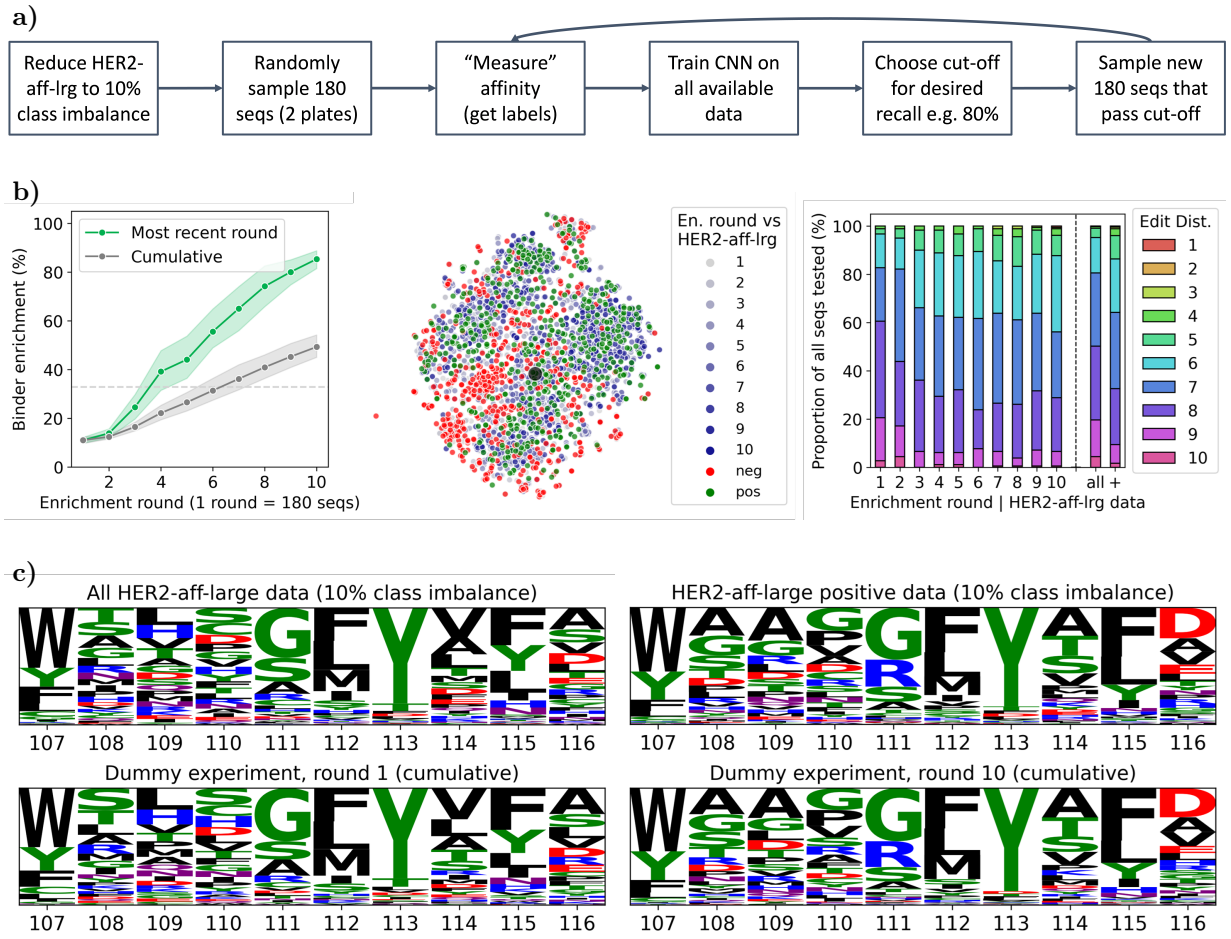

Figure 3.4.7: Results from a simulated experimental setup showing the iterative increase in binder enrichment achieved as more data is collected. Flow chart **a)** illustrates the main steps of this pipeline. First, 180 antibodies (∼two 96-well plates, allowing for controls) were randomly selected from our HER2-aff-large dataset. We used a reduced form of HER2-aff-large for this experiment with a class imbalance of 10% (HER2-aff-large-10%) to simulate the lower enrichments one might achieve when using a computational library design method, such as BLOSUM, in place of DMS. After sampling our initial sequences, we 'measured' their affinity i.e. we unblinded their labels in our simulation. With this newly labelled data, we trained a CNN to classify binding from non-binding sequences, using an 80-20 train-validation data split. The trained CNN was then used to screen newly drawn data from HER2-aff-large-10%, keeping only sequences that passed an optimised cut-off and that had not been tested in the previous round. The cut-off was designed to maintain 80% recall on our validation set to ensure we did not prematurely limit our sequence search space. With 180 new sequences available, we 'measured' their affinity and retrained the CNN on all 360 labelled sequences. This process was repeated for ten rounds of enrichment so that our final library contained 1,800 unique antibodies. Figure **b), left** shows the increasing affinity achieved after each round of experimentation and retraining of the CNN (ten repeats with 95% confidence shaded confidence intervals). The dashed line shows the 32.8% enrichment achieved following DMS. Figure **b), middle** shows that in addition to the rapid increase in enrichment, we continue to explore the full HER2-aff-large-10% positive sequence space thoroughly in latter rounds of enrichment. Figure **b), right** further highlights that the CNN does not learn to simply favour shorter edit distances and that the edit distance distribution of latter rounds closely matches that of HER2-aff-large-10% positive data. Finally, Figure **c)** supports the above, showing that, as expected, the logo plot of our antibody library following the first round of experimentation matches that of HER2-aff-large-10%. However, after ten rounds of enrichment, the logo plot of our 1,800-member antibody library closely matched that of the positive subset of HER2-aff-large-10%.

3.5 Discussion

Machine learning shows great promise for accelerating therapeutic antibody design and optimisation. The field, however, lacks robust baselines quantifying expected binder enrichments and computational strategies that can be seamlessly integrated with experimental workflows.

To support baseline studies, we generated the largest publicly available dataset of antibody variants with antigen binding affinity labels (>524,000 Trastuzumab CDRH3 variants). ML models trained on this dataset were shown to accurately distinguish high- from medium- and low-affinity HER2-binders. A CNN architecture struck the balance between simplicity and complexity: it outperformed a more complex, structure-based EGNN model when trained on small datasets and was more robust to strict train-test splits than simpler, tree-based models. ML classifiers with low training data requirements, such as our CNN, are useful when obtaining large amounts of labelled data is too costly or impractical. Additionally, these classifiers could be used to screen and prioritise novel designs in an iterative development cycle.

We also explored the effectiveness of computational methods (BLOSUM, AbLang, ESM, and ProteinMPNN) for novel library design. A large proportion of sequences generated using each method were predicted to bind with high affinity. The predicted enrichments of 19-30% are close to those achieved by traditional DMS approaches, of ∼33%. Experimental validation of our predictions, which will continue to remain essential, is ongoing. Predicted binders exhibited broad sequence diversity, although not always within a method. Benefits could therefore be found by employing multiple methods in the initial stages of library design if maximising diversity is a top priority.

Finally, we combined the above concepts in a simulated iterative development cycle. Starting with just 180 affinity-labelled sequences with an assumed 10% minimum true enrichment of our computationally designed libraries, we simulated iteratively training the CNN, scoring sequences, and experimentally validating designs. Binder enrichment ratios were found to increase rapidly, surpassing traditional DMS approaches after just three rounds of training and testing. After ten rounds (fewer than 2,000 sequences) enrichments exceeded 80%. The resulting

library closely resembled the high-affinity variants in the HER2-aff-large dataset in terms of sequence identity, edit distances, and diversity. These similarities indicate that our approach continues to fully explore the entire high-affinity sequence space, but does so more efficiently by avoiding testing low-affinity designs. This iterative computationally-aided cycle is fast and independent of the target or library design method, benefitting both animal immunisation and antibody phage display discovery campaigns.

Recent breakthroughs in deep learning, such as generative AI, promise revolutionary changes to the world of drug discovery. We demonstrate, however, that comparatively simple methods may achieve similar performances. Effectively integrating lightweight *in silico* and experimental methods into a continuous learning cycle will allow high-affinity antibody variants to be explored at low costs and timescales in most research settings.

In the next chapter, we adapt our lightweight CNN architecture introduced here and use it to classify whether antibodies are human or non-human. In this new application, we look beyond the CDRH3 and search the entire F_v region for potential immunogenic epitopes.

4 | Humatch - fast, gene-specific joint humanisation of antibody heavy and light chains

4.1 Abstract

This chapter introduces 'Humatch', a computational tool designed to offer experimental-like joint humanisation of heavy and light chains in seconds. Humatch consists of three lightweight Convolutional Neural Networks (CNNs) trained to identify human heavy V-genes, light V-genes, and well-paired antibody sequences with near-perfect accuracy. We show that these CNNs, alongside germline similarity, can be used for fast humanisation that aligns well with known experimental data.

The following text is adapted from *Humatch - fast, gene-specific joint humanisation of antibody heavy and light chains* (Chinery et al. 2024b).

4.2 Introduction

The antibody drug discovery process is a challenging, multi-objective optimisation problem. This problem requires the development of potential leads that bind their target strongly (high affinity), have few off-target effects (high specificity), and possess good developability characteristics (Wang et al. 2021).

One critical step in the development of antibody therapeutics is humanisation (see Section 1.2.4.3). Humanisation is important as in many instances, drug precursors originate in animal models. Gordon *et al.* (Gordon et al. 2024) found that approximately 60% of therapeutics listed in Thera-SAbDab (Raybould et al. 2020) are not genetically human in origin and that this percentage has remained constant over the past two decades.

In humanisation workflows, animals, such as mice, are exposed to the antigen of interest, an immune response is raised, and dominant clones are obtained through library screens (Williams et al. 2010). These clones, which constitute precursor therapeutics, have binding sites optimised to bind the target antigen. However, the rest of the antibody, predominantly the framework region (FR), could contain human immunogenic epitopes. These epitopes risk raising anti-drug antibody (ADA) responses in human patients (Hwang et al. 2005). It is therefore critical to mutate these regions before starting human trials whilst maintaining strong binding and high expression.

Classical humanisation techniques may involve grafting antigen-specific Complementarity Determining Region loops (CDRs) onto a human antibody framework and back-mutating Vernier zone residues (named after a 'vernier scale' because they fine-tune CDR positions (Foote et al. 1992)) to the precursor sequence (Jones et al. 1986; Riechmann et al. 1988). Alternatively, iterative mutations towards a target human germline are made, typically on surface-accessible residues, with these optimised through experimental trial and error (Pedersen et al. 1994; Roguska et al. 1994). These classical approaches can succeed in humanising precursor sequences but are time and cost-intensive, require many mutations that could disrupt binding, and may still lead to therapeutics with high ADA levels (Marks et al. 2021). Recently, computational

tools have been developed to aid in this process.

Hu-mAb (Marks et al. 2021) is one such computational tool that consists of many human gene-specific random forest (RF, see Section 1.3.4.1) classifiers. These RFs were trained on data from the Observed Antibody Space (OAS) (Kovaltsuk et al. 2018; Olsen et al. 2022b) database and succeeded in identifying human heavy and light V-genes with near 100% accuracy on Hu-mAb's test set. Hu-mAb humanises heavy and light chains separately by making all possible single-point mutations to a starting sequence, scoring them with its RF models, selecting the top variant, and repeating until a target threshold is met. Hu-mAb is widely used, however, the humanisation process is slow (∼18 minutes) and susceptible to getting stuck in local minima. Furthermore, its classifiers were only trained on species where OAS contained significant sequence data and some human V-genes are missing. CDR mutations are also strictly forbidden.

BioPhi (Prihoda et al. 2022) is an alternative platform consisting of OASis, a humanness classifier and Sapiens, a transformer-based (Vaswani et al. 2017) humanisation tool (see Section 1.3.4.4). OASis compares all possible 9-mers in an input sequence to how often each is observed within a set of human sequences from OAS. 9-mers that are observed frequently are ranked highly, while those rarely observed receive low scores. Sapiens humanises sequences towards high OASis scores and is trained solely on OAS human sequences using a masked language model approach. During humanisation, probabilities for each residue position are calculated and the amino acids with top predictions are selected (CDR residues are ignored). This process is repeated up to four times. Given all probabilities are calculated in one single pass, humanisation is fast. However, Sapiens lacks the option to select the desired germline; instead, this selection is implicit during training. Additionally, OASis' 9-mer peptide scoring system can rank sequences that are 'between' genes highly e.g. starting HV1 (heavy V-gene 1) and ending HV3. Finally, like Hu-mAb, Sapiens optimises heavy and light chains independently. This separate humanisation risks disrupting expression, lowering stability, and creating immunogenic epitopes that span both chains (Bancroft et al. 2019; Seydoux et al. 2021).

AbNatiV (Ramon et al. 2024), like Sapiens, was trained with masked unsupervised learning on human antibody sequences from OAS. Three antibody (and one nanobody) models were

trained, each a vector-quantised variational auto-encoder (consisting of sequence encoder and decoder elements, similar to transformers), for heavy (HV), kappa (KV), and lambda (LV) sequences (kappa and lambda are different types of human light chains). Nativeness scores (masked likelihoods) were then calculated per residue and averaged to provide an overall nativeness score. This nativeness score can be used to classify human from non-human sequences with high accuracy. The humanisation process is similar to Sapiens, though only mutations to residues observed in its human training data (calculated using a position-specific scoring matrix, PSSM) are allowed by default. Mutations are also limited to solvent-accessible framework residues. This step seeks to avoid unnecessary mutations to residues that cannot form immunogenic epitopes, however, the structural modelling required increases runtimes. To lessen this time, a single model of the wildtype structure is used for all variants but this model risks becoming unrepresentative as the number of mutations increases. Also, the PSSM used during humanisation is not germline-specific, potentially allowing mutations to hybrid mixed genes, similar to Sapiens. AbNatiV, like Hu-mAb and Sapiens, humanises heavy and light chains independently.

In this chapter, we aim to build and improve upon previous work by designing a fast and accurate human classification and humanisation tool, Humatch. The humanisation logic of Humatch produces sequences that align well with experimentally optimised sequences. Humatch is also the first humanisation tool to jointly humanise heavy and light chains with the goal of maintaining high expression, good stability, and removing potential immunogenic epitopes formed across the two chains. Furthermore, Humatch is designed to consistently push designs towards single human V-genes at all stages, avoiding potential hybrid mixed gene designs. Humatch is available open-source and can be easily integrated into existing development pipelines.

4.3 Materials and Methods

4.3.1 Train-test dataset creation

Humatch consists of three CNNs trained to identify (1) human heavy V-genes, (2) human light V-genes, and (3) well-paired antibody sequences. These CNNs were trained on data from the Observed Antibody Space (OAS) database (Kovaltsuk et al. 2018; Olsen et al. 2022b). Both unpaired and paired sequences were processed to remove redundancy. Sequences lacking Cystines at IMGT positions 23 and 104 were also removed. Finally, we required that both the first (IMGT position 1) and last (IMGT positions 128 and 127 for the heavy and light chains respectively) residues were present. As Humatch has been trained on complete VH and VL sequences, users should only input complete sequences for reliable results.

Table 4.3.1 shows the total number of heavy and light sequences used to train Humatch's unpaired CNNs, broken down by V-gene. We did not split our data by D- or J-genes, so each V-gene class includes a mix of these other genes. In total, 8.26m human heavy sequences and 12.73m human light sequences were used for training, alongside 3.77m and 1.41m non-human heavy and light sequences, respectively. This data was subject to a stratified 80-10-10

	Heavy	Lambda	Kappa
V1	2,107,242	1,100,881	3,073,544
V2	263,277	2,266,529	974,970
V3	2,949,663	1,112,650	2,878,296
V4	2,436,892	72,589	867,144
V5	386,538	27,716	9,065
V6	35,566	107,905	46,063
V7	84,216	114,900	4,122
V8	-	36,861	-
V9	-	21,353	-
V10	-	13,678	-

Table 4.3.1: Total number of human sequences used to train, validate and test Humatch's heavy and light chain CNN classifiers, broken down by V-gene. Heavy and Kappa V-genes 8-10 do not exist in OAS. Lambda and Kappa sequences were used together to train CNN-L.

train-validation-test split.

Humatch's paired CNN was trained on 1,673,734 paired human sequences from OAS, and 5,009,443 artificially 'badly' paired human sequences. Both datasets were processed and split similarly to our unpaired data. Previous studies (Goldstein et al. 2019) have found that most 'shuffled' VH-VL pairings of antibodies specific to one target continued to express. However, further experiments were not conducted to determine whether the stability of those that continued to express was reduced. To increase the chances of our artificially paired sequences being poor pairs, we paired only unpaired heavy and light chains from different B-cell classes (e.g. memory vs naive) and different OAS studies. We also ensure that the distribution of true and artificial heavy-light V-gene pairings are similar, ensuring the paired CNN cannot simply learn to identify unusual pairings (see Appendix).

IMGT numbering for all sequences was taken from OAS, originally numbered using ANARCI (Dunbar et al. 2016a). Sequences we padded and aligned to the 200 most common sequence positions identified by KASearch (Olsen et al. 2023), more than the 152 positions used previously by Hu-mAb (Marks et al. 2021). IMGT positions not present in a sequence were represented with pad tokens. Paired heavy and light chains were joined with a length-ten pad, creating a length 410 sequence. Sequences were encoded using ten-dimensional physio-chemical Kidera feature vectors (Kidera et al. 1985) and pad tokens were represented with zero vectors.

Antibody structural data was not used by Humatch given the sparsity of human and non-human crystal structures (see Section 1.3.2.2) and the fact most sequence training data from OAS is unpaired, limiting the usefulness of structural modelling tools.

4.3.2 Humatch Convolutional Neural Network classifiers

In this chapter, we refer to Humatch's three CNNs as CNN-H, CNN-L, and CNN-P for heavy, light, and paired classification respectively.

All three of Humatch's CNNs included 40 convolutional filters, each with a kernel size of ten and a stride of one. The models' inputs were encoded with ten-dimensional Kidera feature vectors and padded to be of size (batch size $\times$ sequence length $\times$ 10). The sequence length was 200

for unpaired sequences and 410 for paired sequences. When passed through the convolutional filters, the input was padded with zeroes at the start and end to ensure the filter outputs were the same length as the input. ReLU activations were used throughout.

Dropout layers with probabilities of 20% were applied following each convolutional filter. These outputs were then max-pooled with a pool size of two and a stride of one. Padding was not applied at this stage, so the lengths of the outputs were one less than the inputs. These outputs were then flattened.

Probabilities of the heavy and light chains being human and well-matched were obtained by passing the output above through two dense layers. These dense layers reduced the output dimensions first to 300 and then to the number of label classes for each CNN. Humatch's heavy CNN classifier has an output dimension of eight (one non-human class and seven heavy V-genes), the light CNN has an out dimension of 18 (one non-human class, ten lambda V-genes, and seven kappa V-genes), and the paired CNN has an output dimension of two (fake and true pairs). A sigmoid activation function is applied to the final layer to ensure predictions across all classes sum to one.

Training of all models used Adam optimisation, binary cross-entropy loss, a learning rate of 7.5×10^{-5}, and a batch size of 1,024. Each model was trained for a maximum of 15 epochs. Weights were saved following each epoch and optimum weights were selected to ensure both high classification accuracy and smooth humanisation. This selection was manual - validation set precision and recall values were examined and example sequences were humanised using different combinations of CNN-H, CNN-L, and CNN-P weights.

Humatch's CNN and training code used Python v3.9 TensorFlow v2.16, and Keras v3.0. Details of all dependencies used can be found at github.com/oxpig/Humatch.

4.3.3 Humatch humanisation logic

Humatch uses a combination of rapid germline lookups and CNN predictions to guide its humanisation process. All steps of this process are described below and summarised in Figure 4.3.1. If at any time the sequence meets all three CNN humanisation target thresholds (e.g.

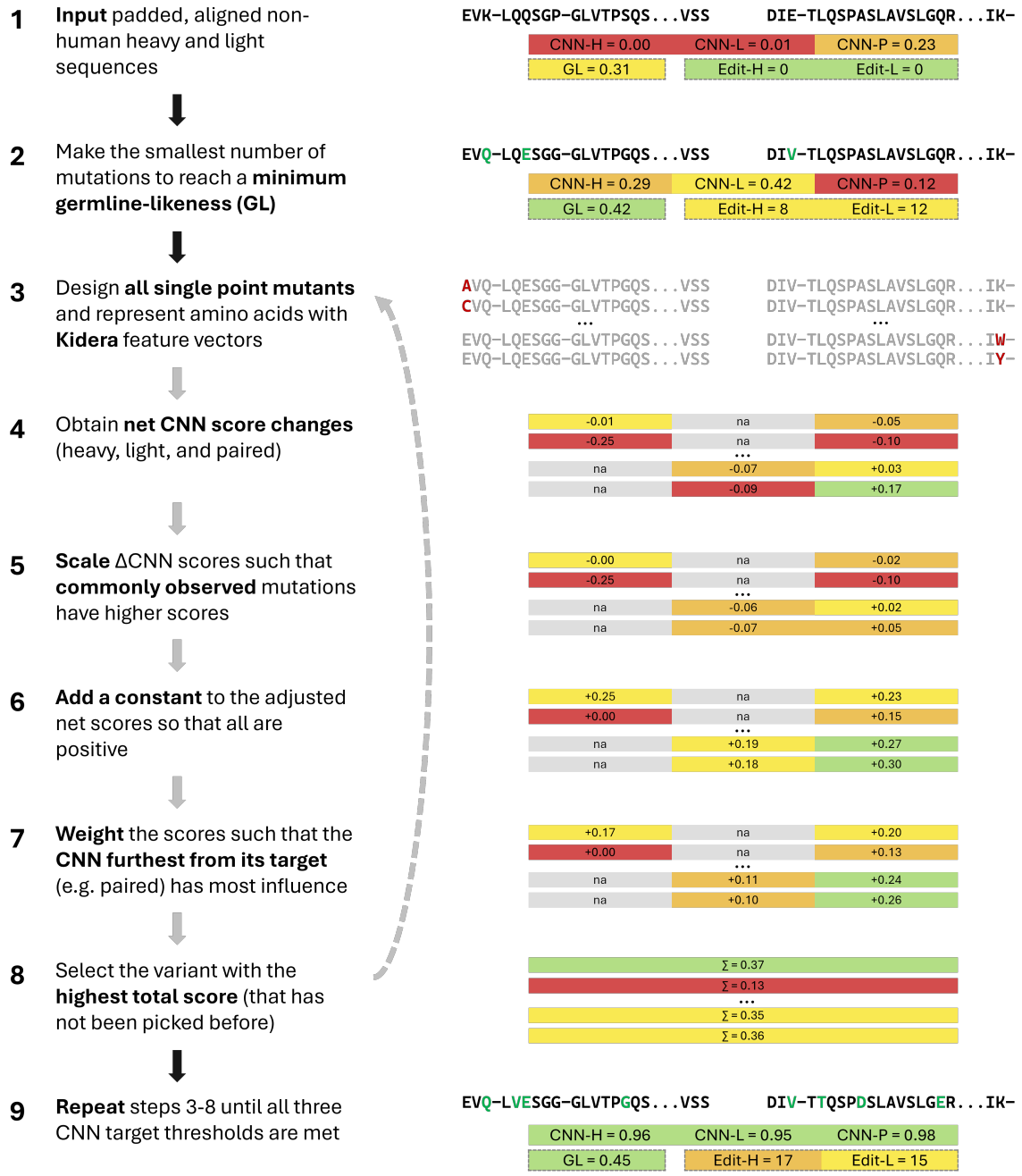

Figure 4.3.1: An overview of Humatch's humanisation procedure including example sequences, CNN scores (H=heavy, L=light, P=paired), germline-likeness scores (GL), and heavy and light precursor edit distances. All numbers are for demonstration purposes only. If at any point during the humanisation process all CNN target thresholds (default 0.95) are met the humanisation process exits. Targeting an initial germline-likeness (step 2) is fast and places Humatch on a sensible humanisation trajectory. All possible single-point variants are designed in step 3, but only four are included in the figure for clarity. Users may select residues to remain unmutated throughout the process (CDRs are ignored by default). In step 8, if the highest-scoring sequence has been selected before, the next highest-scoring novel sequence is selected, avoiding Humatch becoming stuck in local minima. The humanisation process can exit early if a total maximum allowed edit distance is reached (default 60).

0.95 for the heavy, light, and paired CNNs) then the humanisation process stops. A maximum edit distance from the precursor sequence (e.g. 60) can also be set to break the humanisation process early.

The humanisation process starts by calculating a 'germline-likeness' score for the input sequence. This score measures how much the sequence looks like the positive training data of the target gene (chosen by the user or selected automatically). Specifically, Humatch precomputes how frequently each of the 20 canonical amino acids is observed at every sequence position e.g. for HV1, 'Q' is found at IMGT position one 95.8% of the time, while 'E' is observed there 4.0% of the time. These frequencies are extracted for each residue in the input sequence and the mean is taken (padded positions receive zero scores). This germline-likeness score is then compared to a target value, with a default of 0.40 for both heavy and light chains. This cutoff was determined based on the lower 20th percentile of germline-likeness scores achieved by 744 approved and phase 1–3 therapeutics obtained from Thera-SAbDab (August 2024) and Marks *et al.* (see Figure 4.3.2). If the current score is below the target, mutations are made based on whichever will lead to the single largest increase in the germline-likeness score. This process happens separately for the heavy and light chains and continues until the thresholds for each chain are met. No mutations are made at this stage if a chain's germline-likeness score already meets the target threshold.

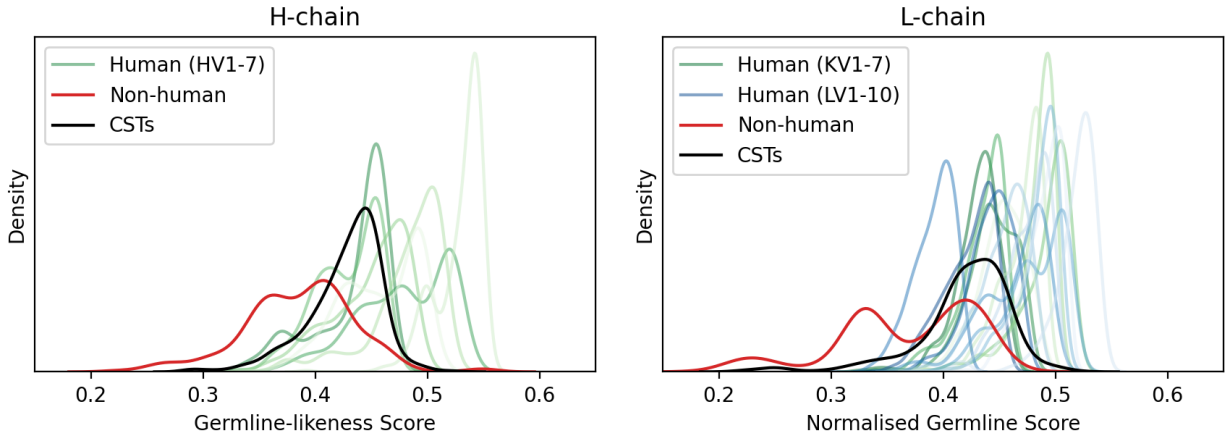

Figure 4.3.2: Comparison of the germline likeness scores for a random selection of heavy and light chain sequences from OAS and 744 clinical-stage therapeutics (CSTs). As expected, on average we observe that human sequences (green and blue, lower number genes are darker) have higher human germline-likeness scores than non-human sequences (red). Therapeutic sequences (black) tend to have high germline-likeness scores, comparable to human sequences.

After this initial quick germline shift of the input sequence, Humatch makes every possible single-point mutation (ignoring padded positions) and scores these with its three CNNs. Heavy chain mutations will affect the predictions of CNN-H and CNN-P only, while light chain mutations will impact CNN-L and CNN-P. CDR mutations are excluded by default in this step (this reduces the likelihood of disrupting binding and speeds up humanisation) though users can add or remove residues from consideration as desired. Once predictions have been made, net predictions are calculated for each CNN separately by subtracting the scores achieved by the unmutated sequence (the output of the germline-likeness step and Humatch's current 'best' design).

The net predictions for each CNN are then scaled using the same germline-specific frequencies from step one. Each positive net prediction is multiplied by the observed frequency of the single-point mutation made. Negative net predictions are multiplied by one minus the frequency. The effect of this step is to prioritise predictions that both increase humanness (separately for each CNN) and are frequently observed. We scale the net predictions rather than the absolute probabilities so that mutations that improve a CNN score always rank above those that worsen it. 'Noise' can be added at this stage by adding a fixed number (as opposed to e.g. Gaussian noise) to each observed frequency before scaling. Adding more noise reduces how much the humanisation process favours previously observed mutations. By default, Humatch adds a low noise value of 0.01.

Once scaled, the net predictions are then made positive again by subtracting the lowest negative score across all three models from all adjusted net predictions. Choosing to add this fixed value, instead of re-adding the predictions of the best sequence, equalises the importance of all three CNN scores ahead of the next stage.

These positive predictions are then scaled by how far away each current CNN score is from its respective target before they are combined e.g. if the current sequence has a heavy CNN score of 0.10, a light CNN score of 0.97, and paired CNN score of 0.55, with targets of 0.95 for all three models, then the heavy predictions would be multiplied by 0.85, the light by zero (as the threshold is already met), and the paired by 0.40. The CNN-H and CNN-L predictions

are then concatenated and added element-wise to the CNN-P predictions. This second scaling stage before summing ensures Humatch does not waste mutations optimising one trait beyond the target threshold while others remain low.

Once summed, the top-ranked variant (that has not been selected in previous rounds) is chosen as Humatch's new best sequence. If this new sequence surpasses all three CNN thresholds or the maximum allowed edit distance is reached, the humanisation process stops. Otherwise, the humanisation process, excluding the initial rapid germline-likeness matching, repeats.

Note that in the above example, as CNN-P considers both heavy and light mutations, a light chain mutation could still be selected despite the CNN-L threshold being met. In this instance, the CNN-L score of the newly selected sequence could differ from the previous best sequence (positively or negatively). If the light chain CNN score were to drop back below the target threshold following this, then the CNN-L scores would be considered again in the next round of mutations.

4.4 Results

4.4.1 Humatch classifies human heavy, light, and naturally paired sequences with high accuracy

When trained on aligned sequence data from OAS (see Methods), Humatch's heavy and light CNN classifiers (CNN-H & CNN-L) separate all human V-genes from one another and non-human sequences with high accuracy (see Table 4.4.1). Some V-genes, such as KV7, have lower accuracy than other classes (PR AUC of 0.85 vs 0.99+ for all other classes) due to a lack of training data (4,000 sequences vs 3m+ for the most populous class, see Methods). In the Appendix, we show Humatch maintains high accuracy even when the training and test data is split by allele.

Other humanness classifiers, such as Hu-mAb, report higher accuracy metrics. However, Hu-mAb's training data did not include rhesus sequences (which are harder to separate from human sequences than mouse from human), or the data sparse V-genes LV9 and KV7. Humatch can achieve comparable accuracy to Hu-mAb when trained and tested on the same data (see Appendix). To prevent overfitting and allow for a smoother humanisation process, we stopped our CNN training early after three epochs for both CNN-H and CNN-L.

In addition to unpaired heavy and light chain classifiers, Humatch includes a third CNN (CNN-P) trained on naturally paired human heavy and light chains and artificially 'badly' paired sequences. Training of CNN-P was also stopped early after 15 epochs to prevent overfitting. Despite potential noise in the training data that arises from the use of artificial bad pairing (see Methods), Humatch can accurately separate naturally paired from artificially 'badly' paired human sequences (PR AUC and ROC AUC above 0.99, see Table 4.4.1).

The outputs of all three CNNs are made available for paired VH/VL humanness classification, though the unpaired models can be run individually. Default thresholds of 0.95 are recommended for all classifiers to ensure high precision and recall (see Appendix). These thresholds can all be adjusted by users.

Class	PR AUC	F1	MCC	ROC AUC	BA
HV1	0.998	0.971	0.966	1.000	0.991
HV2	1.000	0.991	0.991	1.000	0.996
HV3	0.999	0.983	0.977	1.000	0.992
HV4	1.000	0.987	0.983	1.000	0.995
HV5	1.000	0.987	0.987	1.000	1.000
HV6	0.992	0.911	0.914	1.000	0.994
HV7	0.992	0.897	0.900	1.000	0.990
LV1	1.000	0.998	0.998	1.000	1.000
LV2	1.000	0.995	0.994	1.000	0.999
LV3	1.000	0.997	0.997	1.000	1.000
LV4	1.000	1.000	1.000	1.000	1.000
LV5	0.999	0.959	0.960	1.000	1.000
LV6	1.000	1.000	1.000	1.000	1.000
LV7	0.999	0.982	0.982	1.000	0.999
LV8	1.000	1.000	1.000	1.000	1.000
LV9	1.000	0.998	0.998	1.000	1.000
LV10	0.999	0.966	0.966	1.000	0.998
KV1	1.000	0.983	0.978	1.000	0.994
KV2	1.000	0.988	0.987	1.000	0.999
KV3	1.000	0.993	0.992	1.000	0.998
KV4	1.000	0.998	0.997	1.000	1.000
KV5	1.000	1.000	1.000	1.000	1.000
KV6	1.000	0.994	0.994	1.000	1.000
KV7	0.848	0.587	0.642	1.000	0.993
true pairs	0.995	0.966	0.954	0.998	0.983

Table 4.4.1: Performance of Humatch's three CNN classifiers - heavy, light, and paired. Sequences belonging to all genes are classified with high accuracy. Gene classes with the lowest scores tend to have little training data available. PR AUC = Area Under the Precision-Recall Curve; F1 = F1-score, MCC = Matthews Correlation Coefficient; ROC AUC = Area Under the Receiver Operating Characteristic Curve; BA = Balanced Accuracy.

4.4.2 Humatch's CNN-P scores correlate with thermostability

Humatch's CNN-P classifier is designed to ensure that the variable heavy (VH) and variable light (VL) regions remain well-matched during the humanisation process. This pairing consideration aims to maintain good expression and stability and account for the fact that immunogenic epitopes can be formed across the two chains (Bancroft et al. 2019; Seydoux et al. 2021).

To demonstrate the utility of including a paired model we examined the relationship between high CNN-P predictions and thermal stability using 137 therapeutics with melting temperature data from Jain *et al.* (Jain et al. 2017) (see Figure 4.4.1). In total, 98 therapeutics received high scores (CNN-P > 0.5) and 39 had low scores (CNN-P ≤ 0.5). While expression, measured using HEK titers, showed no significant correlation according to a two-sided T-test ($p = 0.13$), higher CNN-P scores were shown to correlate with greater melting temperatures ($p = 0.00092$).

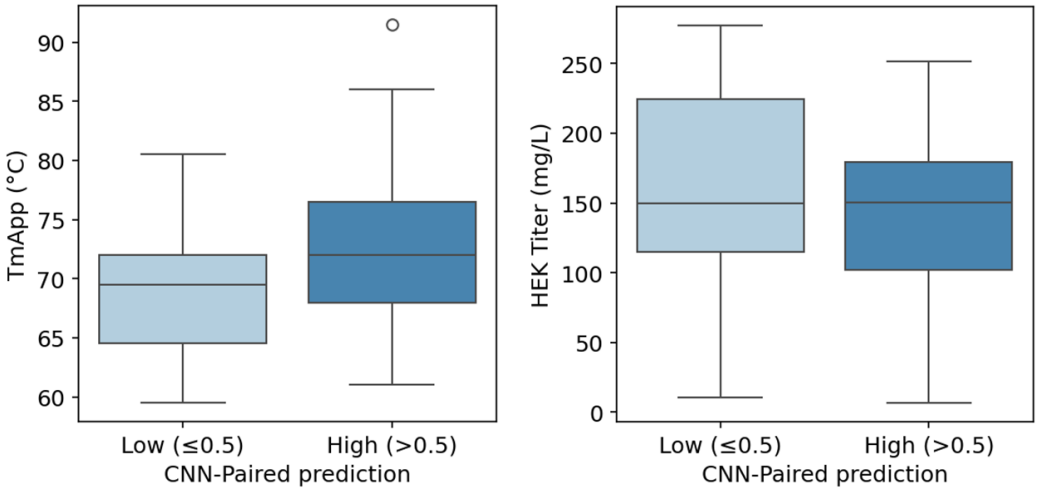

Figure 4.4.1: Box plots showing the melting temperatures (TmApp, left) and expression levels (HEK titers, right) of 137 therapeutics from Jain *et al.* grouped by Humatch's CNN-P scores.

4.4.3 Humatch identifies therapeutics that are more likely to have high anti-drug antibody responses

Marks *et al.* introduced a dataset of 217 therapeutics where the anti-drug antibody (ADA) levels had been recorded and corresponding sequence data was found for 211 of them. These sequences were numbered and aligned using ANARCI and Humatch was then used to obtain heavy, light, and paired predictions for all 211 therapeutics.

The minimum score of the three CNNs was calculated, reasoning that if one of the CNNs ranked a therapeutic poorly, then it should be classed as higher risk. The minimum of the CNN predictions was found to correlate with ADA response (see Figure 4.4.2), with the majority of highly immunogenic therapeutics having low Humatch predictions. Using a loose Humatch cut-off of 0.01 to retain most low ADA therapeutics removed 89% of high ADA (red), 59% of medium ADA (yellow), and just 23% of low ADA (green) therapeutics. Alternative humanness classifiers, including Hu-mAb, AbNatiV, and OASis, offer similar separation of low and high ADA antibodies (see Appendix). Better separation might require considering the contributions of non-variable region antibody residues and other compounds that complete the antibody

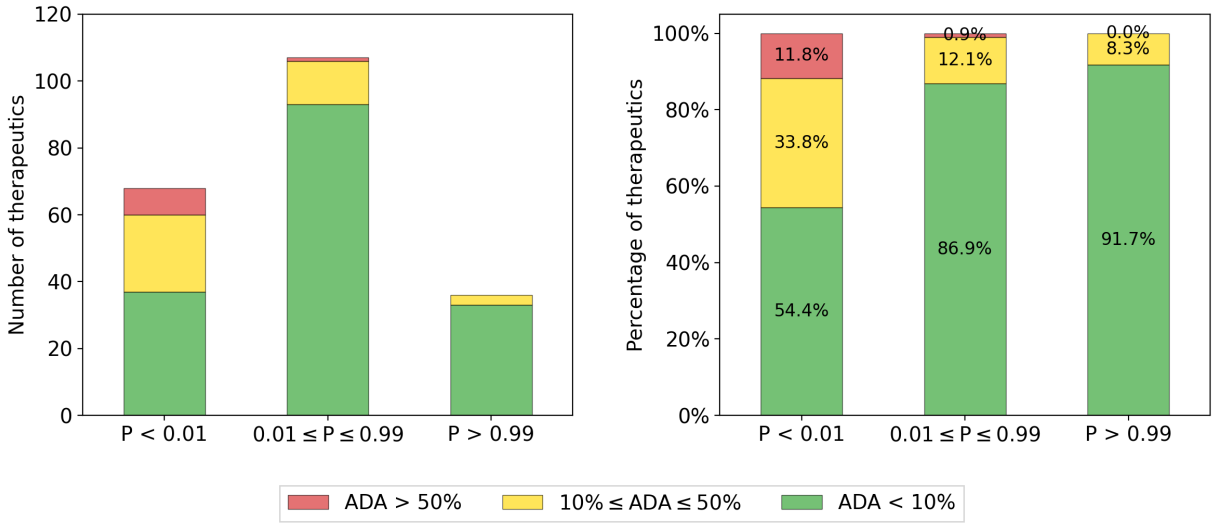

Figure 4.4.2: Categorisation of 211 therapeutics with anti-drug antibody labels by Humatch. The 211 therapeutics were scored by Humatch's heavy, light, and paired CNNs. The minimum of these three scores (P) was used to group the therapeutics into three bins - those with a minimum prediction above 0.99, below 0.01, or between these values. The left plot shows the total number of therapeutics that were grouped in each bin and the right plot shows the proportions broken down by bin. The majority of therapeutics with the highest ADA levels (>50%) fall in the lowest prediction bin ($P <$ 0.01).

drug formulation to ADA responses. Further breakdowns of ADA responses for Humatch's three classifiers can also be found in the Appendix.

4.4.4 Humatch-humanised sequences align well with experiments

Humatch is primarily designed to offer experimental-like humanisation in seconds. This humanisation is achieved through a combination of rapid gene-specific germline-lookup tables and guidance by Humatch's three CNNs. Figure 4.3.1 shows an overview of all the main steps involved in the humanisation process and these are summarised below. Further details are provided in the Methods.

The first step in the humanisation process involves the most common germline mutations being made to the input sequence to achieve a minimum germline-likeness (GL) score of 0.40 for both heavy and light chains (see Methods and Appendix). This initial GL-matching step is fast and allows the humanisation of a starting sequence to any target gene. Next, all single-point variants of the GL-mutated sequence are designed and scored with Humatch's three CNNs. These scores are then scaled to up-weight common germline mutations and prioritise improvements to the lowest CNN score (see Methods and Appendix). The top-scoring variant at this stage is selected and this process is repeated until all three CNN target thresholds of 0.95 are met. In successive iterations, Humatch ignores previous top-scoring variants to avoid local minima. Target GL scores of 0.40 and CNN thresholds of 0.95 were decided based on observed therapeutics GL scores (see Figure 4.3.2) and desired CNN precision-recall values (see Appendix).

To test Humatch we used it to humanise a set of 25 precursor heavy and light sequences with known experimentally humanised therapeutic endpoints (Marks et al. 2021). The target scores were set for Humatch following a similar process to Hu-mAb, using the higher of 0.95 or the scores achieved by the experimentally optimised sequences (multiplied by 0.9999 to avoid forcing CNN scores to 1). All sequences reached their humanisation target scores in fewer than 60 edits (the maximum observed experimentally).

We also conducted a baseline experiment where only the top germline mutations were made to test the importance of Humatch's CNN guidance in the humanisation process. In this

Method	H overlap	L overlap	H edit	L edit	Time (s)
Hu-mAb	*0.67*	*0.77*	*14.8*	*10.6*	1,093
AbNatiV	0.57	0.65	**11.5**	**9.3**	218
Sapiens	0.57	0.69	21.3	17.8	**3**
Humatch (ours)	**0.77**	**0.82**	20.6	13.0	*33*
Experimental	1.00	1.00	26.0	19.2	-

Table 4.4.2: A summary of the outputs of four different humanisation tools when tasked with humanising the same 25 precursor therapeutics. Computationally designed sequences were deemed humanised when their respective target scores were met. These designs are compared to the experimentally optimised sequences (bottom). 'H edit' and 'L edit' are the mean edit distances of the humanised therapeutics from their corresponding precursor sequences. 'H overlap' and 'L overlap' described the mean overlap in mutations made computationally compared to those made experimentally. If all suggested computational mutations were made experimentally, the overlap would be one; if none matched, the overlap would be zero. 'Time (s)' is the mean time in seconds required to humanise each therapeutic (both heavy and light chains). The mean time required for experimental humanisation is not known. The '**best**' computational scores are in bold and '*second best*' in italics.

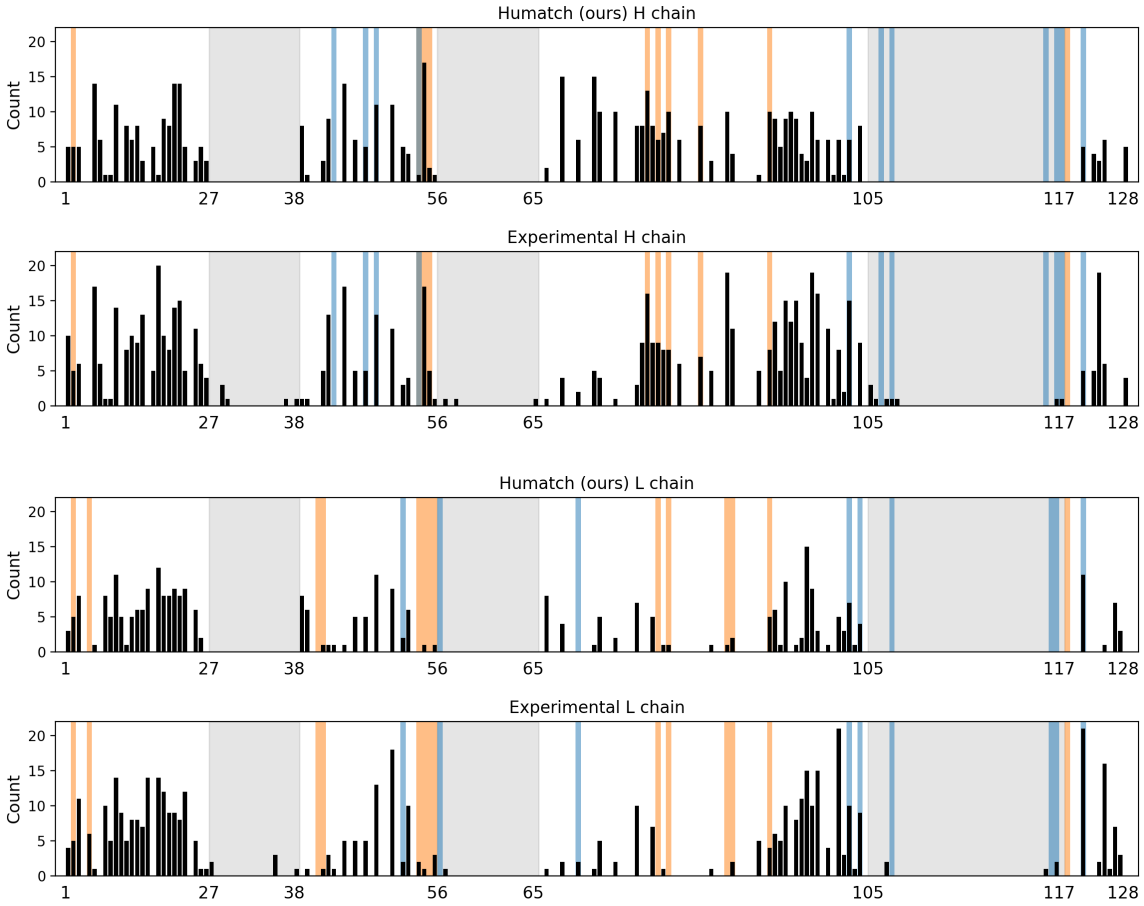

Figure 4.4.3: A comparison of Humatch and experimental mutation profiles for 25 precursor therapeutics, separated by heavy and light chains. The black bars show the number of mutations made at each IMGT sequence position across the 25 therapeutics. CDR regions are shaded in grey, Vernier zone residues in orange, and interface residues in blue.

baseline, the same number of heavy and light chain mutations were made as in Humatch's full humanisation logic. We found that no baseline-designed paired sequences matched Humatch's designs exactly and only two designs passed all three CNN target thresholds of 0.95 (further details can be found in the Appendix).

Table 4.4.2 compares Humatch's humanisation performance to three other antibody humanisation tools - Hu-mAb (Marks et al. 2021), Sapiens (Prihoda et al. 2022), and AbNatiV (Ramon et al. 2024). Hu-mAb's overlap values and edit distances were taken from their paper. All other tools were run with default and paper-recommended parameters (see Appendix). We observed high overlap in the mutations made by Humatch and those made experimentally. Table 4.4.2 shows that, on average, 77% of heavy and 82% of light chain mutations exactly matched between the two, higher than all other methods. The majority of these mutations were to common germline residues. Successful humanisation, according to Humatch's three CNNs, can be achieved in fewer edits by setting a lower initial target germline-likeness score at the expense of slightly smaller overlaps and longer humanisation runtimes (see Appendix). These parameters can be changed by the user when running Humatch as desired.

Figure 4.4.3 shows Humatch's heavy and light chain mutation profiles closely match those from experiments. Vernier zone and interface residues that may affect the stability of the antibody are highlighted in orange and blue, respectively. Humatch made similar proportions of mutations within these zones (19.3% and 12.6% for the heavy and light chains, respectively) compared to those made experimentally (17.9% and 14.6%).

Finally, we observe that when many mutations are made experimentally, Humatch also suggests more mutations and vice versa, with a Pearson correlation of 0.89. This correlation holds when considering the total number of heavy and light mutations and when considering each chain individually (see Figure 4.4.4).

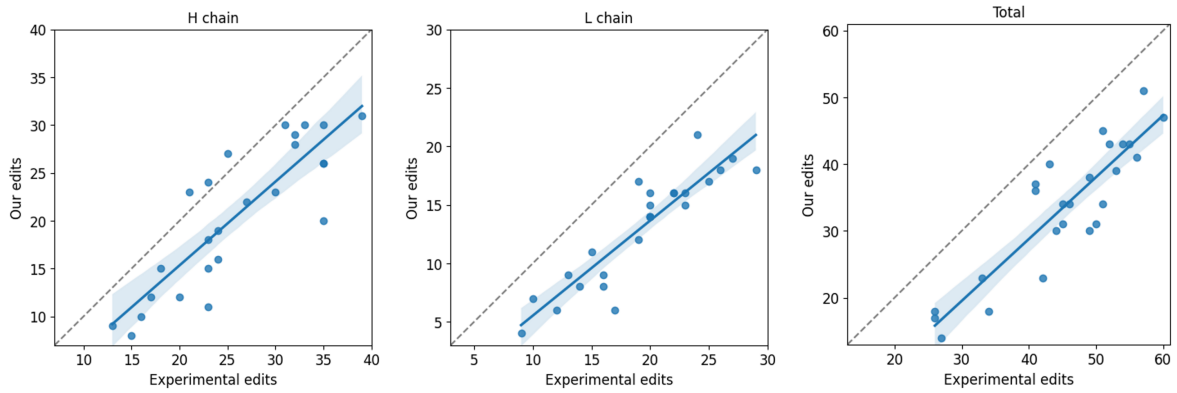

Figure 4.4.4: Comparison of the number of experimental edits made during the humanisation of 25 precursor therapeutics and the number suggested by Humatch. Pearson correlations of 0.86, 0.90, and 0.89 exist for the heavy, light, and total number of edits respectively.

4.4.5 Humatch provides gene-specific humanisation while maintaining good heavy and light pairing

Humatch's architecture and humanisation logic designs sequences with high experimental alignment (see Table 4.4.2). This alignment is achieved while pushing designs towards well-paired specific V-genes and away from other human V-genes and non-human sequences (see Figure 4.3.1 and Methods). Here, we show that other humanisation tools optimise towards different V-genes than those chosen experimentally and that as none consider good VH/VL pairing this can lead to designs that Humatch predicts are unlikely to pair naturally.

Section 4.4.2 showed that high CNN-P scores correlate with greater thermal stability. High CNN-P scores are therefore desirable for stable therapeutic designs. Figure 4.4.5 (top right) shows how Humatch's CNN-P ranks all 25 computational and experimental designs summarised in Table 4.4.2 for their VH/VL pairing.

Hu-mAb's designs are predicted to have the least natural pairing. Some designs from Sapiens and AbNatiV are also predicted to be ill-matched, as are several experimental endpoints (unfortunately, we could only find stability data for 13 of the 20 experimental endpoints predicted to pair well and none for the five predicted to pair less well for validation). The edit distances from each precursor sequence are shown to the right of each plot. Humatch designs VH/VL sequences that are predicted to be well-matched, according to our CNN-P, in fewer edits than were made experimentally for all 25 therapeutics. A lack of ground truth 'badly' paired se-

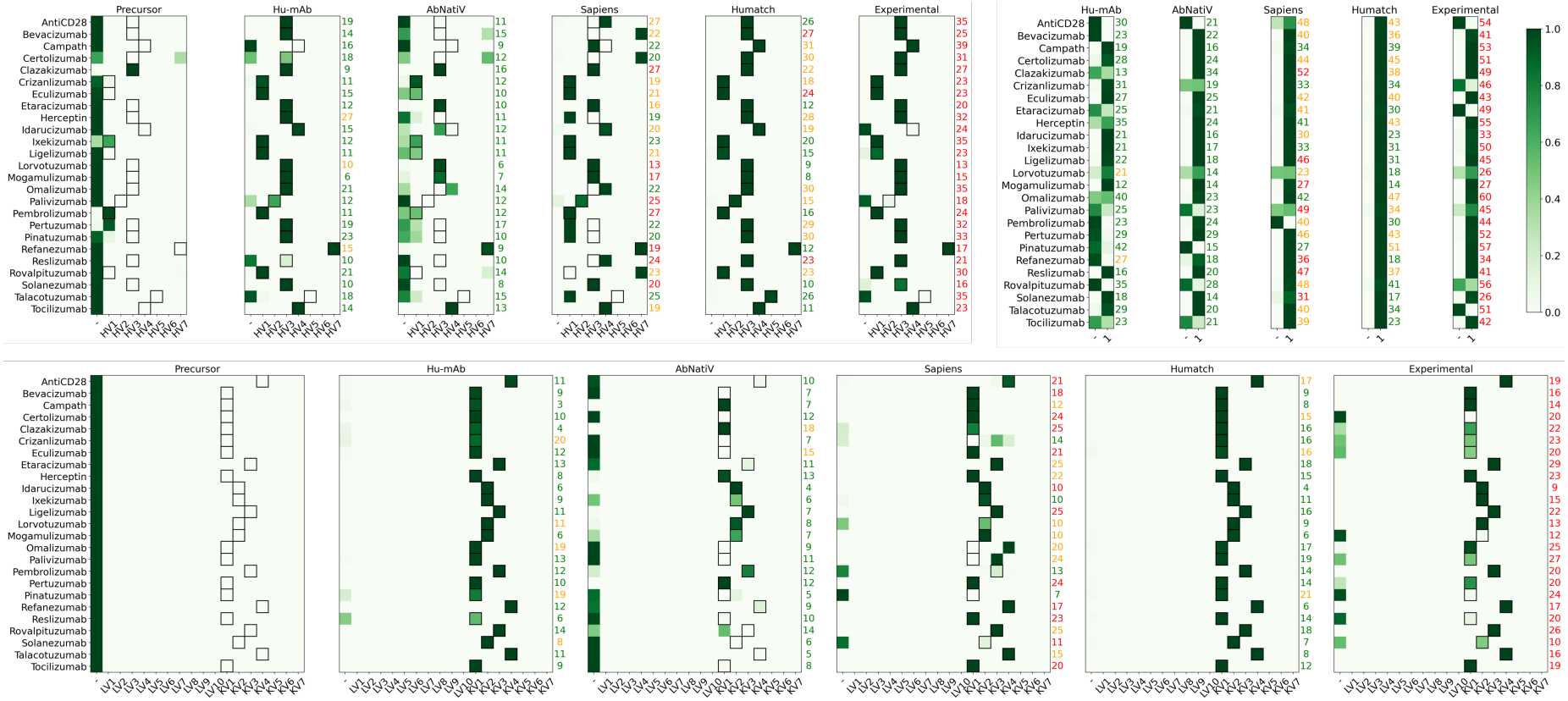

Figure 4.4.5: Comparison of Humatch scores achieved by different tools when humanising 25 precursor therapeutics. Humatch outputs multiclass prediction vectors for the heavy, light, and paired models. The predicted values are coloured according to the colour bar, with higher predictions shown in darker green. The target genes, shown as black boxes, were determined from the experimentally humanised sequences. The black boxes are omitted from the paired model for clarity - in this instance, we target class '1', corresponding to the heavy and light chain being well paired. The edit distances of each design are shown to the right of each plot. Edit distances are coloured according to their relative size compared to experimental edits. Designs with the same or more edits as the experimental are coloured red, designs with fewer than 75% of the number of experimental edits are shown in green, and designs with edits between 75% and 100% of the experimental are shown in orange.

quence data for training CNN-P could result in both false positive and negative predictions; nevertheless, Humatch offers an improvement over ignoring VH/VL pairing concerns during humanisation.

Figure 4.4.5 (top left and bottom) highlights how other humanisation tools - AbNatiV and Sapiens - often target different heavy and light V-genes than the experimentally designed endpoints. Outputs from these tools could also theoretically lie 'between' genes given their humanness scoring protocols. We find no direct evidence of this for these 25 designs (see Figure 4.4.5) but in the Appendix we show that Humatch successfully ranks more artificial 'mixed-gene' designs as non-human (51%) than AbNatiV (28%), Hu-mAb (8%), and OASis (1%).

Despite humanising towards different genes, most designs from Sapiens are still ranked as human by Humatch's CNN-H and CNN-L classifiers. Fewer designs from AbNatiV are ranked as human by Humatch. Disagreement between humanisation tools is common as each may identify different patterns in the training data, even when trained on the same or similar datasets. Like Humatch, other tools also rank some experimentally humanised sequences low - Marks *et al.* find Hu-mAb ranks only ∼66% of experimentally humanised VH/VL pairs as human.

Most of Hu-mAb's designs achieve high CNN-H and CNN-L scores and target the same genes as experiments. In some cases, Hu-mAb, and other tools, achieve these high scores in fewer edits than our Humatch protocol. However, in many of these instances, these designs do not meet the target Humatch score given by the experimental sequence. As mentioned previously, Hu-mAb's designs are also predicted to be less well-paired. If desired, Humatch can design sequences with high CNN-H, CNN-L, and CNN-P scores in fewer edits than stated in Figure 4.4.5 at the expense of lower germline-likenesses (see Appendix).

In summary, Figure 4.4.5 illustrates how Humatch is the only humanisation tool that allows both humanisation towards specific V-genes and good VH/VL pairing.

4.4.6 Humatch humanises sequences rapidly, allowing for high throughput computational design

To test Humatch in a high-throughput environment, we randomly selected 1,000 non-human naturally paired sequences from OAS as potential precursor therapeutics. Humatch scored these starting sequences and the top-scoring heavy and light genes were chosen as target genes during humanisation. Similarly to before, heavy, light, and paired target thresholds were set at 0.95 and the humanisation of each pair of sequences was allowed to continue up to a maximum combined edit distance of 60 from the starting sequence.

As Humatch includes logic to avoid local minima (see Figure 4.3.1 and Methods) it successfully humanised most sequences according to the above criteria, though 12 VH/VL pairs failed to humanise after hitting the maximum edit distance threshold. The mean heavy and light edit distances of the humanised sequences from their precursors were 11.6 and 13.9, respectively. These edit distances are lower than those reported for the 25 therapeutics earlier (see Table 4.4.2), reflecting the higher experimental target thresholds used in some cases. The mean time required for humanisation was 36s when run on a standard desktop computer with 16 CPUs.

4.5 Discussion

Humanisation is a critical step in the design of many antibody therapeutics (Gordon et al. 2024). However, experimental optimisation involves trial and error and is time and cost-intensive (Wang et al. 2021). Computational tools exist to aid in this process, though many are limited by long runtimes, no option to specify target germlines, and a lack of coherent joint heavy and light chain optimisation (Marks et al. 2021; Prihoda et al. 2022; Ramon et al. 2024).

To support the rapid design of low immunogenicity, humanised antibody therapeutics, we have developed Humatch. Humatch is a collection of three CNN classification models that accurately identify human heavy sequences (CNN-H), human light sequences (CNN-L), and well-paired antibody VH/VL pairs (CNN-P). Humatch uses these classifiers and gene-specific germline data to guide its humanisation process through an iterative cycle of designing and ranking all possible single-point variants.

Humatch's heavy and light antibody V-gene classifiers offer wider applications than previous methods as they are trained on a more extensive set of germlines and non-human species, including both human LV9 and KV7 genes and rhesus sequences. Humatch achieves near-perfect classification accuracies for most genes, though we avoid targeting the absolute highest accuracies to prevent overfitting, and instead attempt to favour a smoother humanisation process.

To further smooth the humanisation process, sequences are encoded using Kidera feature vectors. One-hot encodings were trialled and provided similar classification accuracies but designed sequences with lower experimental overlap. We hypothesise that due to their sparse nature and the fact that each convolutional filter acted over the entire sequence, one-hot features allowed 'nonsense' mutations never observed in human or non-human sequences to receive high CNN predictions by chance. Kidera feature vectors provided near-perfect classification accuracies so were selected over more descriptive Large Language Model (LLM) embeddings to restrict the size of Humatch's model weights and runtime.

Humatch's CNN-H and CNN-L classifiers output multiclass predictions (e.g. non-human plus all human heavy V-genes), meaning that during humanisation they push sequences not only

towards their target gene but also away from other human genes. Gene-specific amino acid frequency lookup tables, similar to chain-level PSSMs used by AbNatiV, also help in this guidance, avoiding humanising sequences 'between' genes. Mixed-gene sequences are not observed naturally and so pose potential immunogenic risks, yet we show many are ranked highly by other humanisation tools.

With its CNN-P, Humatch also ensures heavy and light chains remain well paired during the humanisation process. This offers potential expression and thermostability benefits as high CNN-P predictions are shown to correlate with higher melting temperatures. CNN-P also considers the fact that immunogenic epitopes could be formed between heavy and light chains - therapeutics with the highest anti-drug antibody levels have low CNN-P predictions (see Appendix). Some caution should be applied when interpreting these predictions for other tools though given the lack of 100% ground truth 'badly' paired data in OAS. Nevertheless, Humatch is the only humanisation tool that explicitly optimises for well-matched sequences.

Generally, it is difficult to compare the classification and humanisation outputs of different humanisation tools. Even when trained on the same or similar data, various architectures can learn to identify and prioritise different patterns in the data. For this reason, we prioritised high experimental overlap and germline-likeness in optimising Humatch's architecture and humanisation protocol. Humatch's designs overlap with experimental designs more than any other method - 77% and 82% of heavy and light mutations match exactly. This high overlap is achieved despite Humatch making the second-largest number of mutations of all methods, indicating that Humatch does not simply find the 'easy' mutations.

Finally, Humatch offers fast humanisation by targeting an initial germline-likeness score that is calculated based on observed therapeutics. This initial step takes less than a second to complete, immediately sets Humatch on a sensible humanisation trajectory to any target gene, and allows for high throughput experiments with a total runtime of ∼35s per paired VH/VL. Given Humatch's quick runtime, strong experimental alignment, and inbuilt guidance towards single germlines and natural chain pairings, it should speed up and reduce the cost of the drug discovery process.

5 | Conclusions and Future Directions

5.1 Conclusions

This DPhil thesis has introduced new baselines, workflows, and computational tools to aid antibody therapeutic design. Specifically, we sought to address the following three questions: (1) where do antibodies bind, (2) how strongly do they bind, and (3) might they raise an immunogenic response?

To answer the first of these questions, we developed Paragraph - an antibody paratope prediction tool. Paragraph takes modelled antibody structures as input and offers residue-level paratope predictions using graph neural networks. We showed these predictions are faster (given a crystal or modelled antibody structure is available) and more accurate than previous leading tools. Furthermore, Paragraph's performance was found to increase with the quality of the input antibody model structure. This correlation means that Paragraph's accuracy should continue to grow with future improvements in antibody structure prediction without the need for retraining.

Our exploration of antibody binding affinity centred around a new dataset of over half a million Trastuzumab variants. In this study, we showed that convolutional neural networks (CNNs) could identify high-affinity variants after being trained on just a few hundred sequences. This low training data requirement provides evidence that continuous learning could be used in experimental settings to refine and enrich antibody libraries against specific targets. Additionally, when trained on all available data, our CNN achieved near-perfect accuracy. This classifier was used to screen novel libraries designed using BLOSUM, large language models (AbLang

& ESM), and inverse folding methods (ProteinMPNN). All computationally designed libraries were predicted to be enriched in high-affinity variants above a random baseline. Crucially, these methods also designed binders covering areas of sequence space unlikely to be explored using conventional deep mutational scanning approaches. All tools used in this study were already freely available and none were specialised to excel at designing high-affinity antibody libraries. This work therefore provides a robust, sensible baseline for current and future affinity maturation tools to compare against.

Finally, we developed Humatch to better predict and mitigate potential immunogenic responses to antibody therapeutics. Humatch consists of three CNNs that classify whether an antibody's heavy chain is human, whether its light chain is human, and whether the two chains are well-matched. During humanisation, Humatch prioritises mutations towards those commonly observed in the target germline and those that maximise the three CNN scores. This balance during the humanisation protocol leads Humatch to suggest mutations that align well with those decided by experimentalists. Additionally, Humatch is the first tool to explicitly consider heavy-light pairing during humanisation. We show that high Humatch pairing scores correlate with increased antibody stability and suggest it may also help avoid immunogenic epitopes forming between chains.

In conclusion, this thesis provides novel insights and computational tools to aid antibody property prediction and optimisation. Together, these methods can complement traditional experimental techniques, improve their efficiency, and lower total therapeutic development times and costs.

5.2 Future work

With additional time and resources, the projects presented in this thesis could be expanded to improve their accuracy, widen their applications, or provide additional experimental validation of their predictions.

Since Paragraph was published, more antibody crystal structures have been released and the accuracy of antibody modelling tools has improved. Paragraph could be retrained on this larger

dataset with better models to yield incremental improvements. Ideally, we would have also liked to train Paragraph on nanobody and TCR structures. However, the number of antigen-bound crystalised nanobody and TCR structures available was approximately an order of magnitude lower than antibody structures. This comparative lack of data is still true today. In the future, once the number of crystal structures grows to the thousands, Paragraph could be retrained or fine-tuned on these alternative immunoglobulin formats. In the meantime, Paragraph's potential use in guiding computational docking experiments could be explicitly tested, similar to proABC-2's guidance of HADDOCK's docking protocol. Like proABC-2, Paragraph could also be retrained to output multiclass predictions describing which interactions predicted paratope residues are likely to form, such as hydrogen bonds and/or hydrophobic interactions, further constraining docking.

In the coming months, we expect to receive experimental validation of 700 novel Trastuzumab variants designed using BLOSUM, AbLang, ESM, and ProteinMPNN. The subsequent analysis will reveal whether the true enrichments are close to the predicted ranges of 20-30%. This validation will also reveal whether our CNN can accurately predict the binding behaviour of out-of-distribution designs, as many occupied different areas of sequence space to the training data. If the CNN is less predictive out-of-distribution, then alternative feature encodings or network architectures could be explored. Separately, though we were fortunate to have access to a large dataset of Trastuzumab variants for this study, it would be useful to have affinity-labelled data on a wider range of targets with antibody mutations made outside the CDRH3. Such datasets could offer insight into broader antibody-antigen recognition and potentially allow a base model to be trained that could be fine-tuned more quickly in the continuous learning settings suggested earlier.

Humatch could also benefit from experimental validation to build essential trust within the pharmaceutical industry. Unfortunately, only the experimental end-points are known to be sufficiently safe to administer to patients. Our designs, and those from Hu-mAb, AbNatiV, and Sapiens on the other hand are only predicted to be safe. However, validating their immunogenic risks is not simple. Instead, the affinity and melting temperatures of all designs could be tested to first ensure that suggested mutations do not disrupt binding or affect stability. Finally,

though Humatch can humanise sequences in 30 seconds, faster than most other methods, its speed could potentially be improved by using lower-level programming languages. Sequences may also be humanised faster if trajectories requiring fewer edits could be identified. Saliency maps that measure the importance of individual residue features in reaching a CNN classification could be used to identify these nearer endpoints. This would work by identifying the top N most salient residue positions and exhaustively searching all their combinations. Such an approach would be computationally feasible and might result in final designs closer to the original precursor.

Ultimately, work could continue indefinitely to improve and expand upon the research presented in this thesis. Time and this DPhil, however, are limited. Therefore, I hope other researchers find our current results useful and I wish others the best if they pursue some of the suggestions above in the future!

6 | Appendix A for Chapter 3

HER2-aff-large collection methods

The Trastuzumab single chain variable fragment (scFv) CDRH3 dataset used in Chapter 3, HER2-aff-large, was guided by the site-specific Deep Mutational Scanning results generated previously by Mason *et al.* (Mason et al. 2021).

Briefly, a Trastuzumab scFv antibody library was cloned in a pSYD yeast display vector, a variant of the pDNL6 yeast display vector (pSYD uses N-terminal fusion for scFv-aga2 display, while pDNL6 uses a C-terminal fusion of aga2-scFv). The Trastuzumab scFv antibody library cloned in pSYD vector was transformed in EBY100 yeast cells (ATCC #MYA-4941DQ) selected on SD + CAA plates (2% dextrose, 0.67% yeast nitrogen base, and 0.5% casamino acids yeast selection media) at 30°C for 48-72 hours. Yeast display analysis of the Trastuzumab scFv library was performed as described previously by Ferrara *et al.* (Ferrara et al. 2012) and Chao *et al.* (Chao et al. 2006).

The next day, the cell pellet was resuspended in SG + CAA (containing 2% galactose and 0.1% dextrose) at 0.5 OD/ml and incubated at 20°C with shaking for one to two doublings, as determined by OD. The cells were washed with the wash buffer and processed for staining to check HER2 binding. Around 1-10$\times 10^7$ cells were labelled with 100μg/ml anti-V5 tag antibody followed by addition of 100nM HER2 and incubated for 30 minutes on ice. The cells were then washed twice more with wash buffer and labelled with a 1:200 dilution of secondary reagents (goat anti-mouse - Alexa 488 and streptavidin-PE).

Finally, the cells were incubated for 30 minutes on ice, washed twice with a wash buffer, and resuspended in 1ml of sorting buffer. To determine their affinity, the cells were sorted for the brightest V5 FITC positive (scFv expression) antigen binding population (PE positive) and labelled as high-affinity binders, as shown in Figure 6.0.1. The cells were further sorted for the brightest V5 FITC positive medium and low-affinity antigen binding populations. The populations were sorted into tubes containing YPD media and grown in SD + CAA liquid media at 30°C with shaking overnight as described previously (Ferrara et al. 2012).

Plasmid DNA was isolated using a yeast plasmid isolation kit (Zymoprep Yeast Plasmid Miniprep I #D2100) following user protocol. The variable heavy (VH) gene containing the CDRH3 sequence for each population was PCR amplified using in-house NGS-specific primers. The amplicons were PCR-cleaned and prepared for NGS. The DNA libraries were sequenced on Illumina using NovaSeq 6000 S2 Reagent Kit v1.5 (300 cycles) and the raw data has been deposited on Zenodo - doi.org/10.5281/zenodo.10549114. The primers used for generating variable heavy amplicons were:

`NGSVH Fwd: 5´CACCCGTTATGCCGACAG3´`

`NGSVH Rev: 5´GGGATTGGTTTGCCGCTAG3´`

The raw paired NGS reads were merged using PEAR (v0.9.6). The subsequent dataset consisted of 618,585, 799,368, and 663,397 high, medium and low-affinity unique CDRH3 sequences, respectively. Singleton (count=1) sequences were removed from the dataset to improve the quality of the data. The final Trastuzumab variant dataset comprised 178,160, 196,392, and 171,732 sequences in 'high', 'medium', and 'low' affinity binder classes, respectively. The heavy and light chain sequences (from *1n8z*) were numbered according to the IMGT scheme. Heavy chain insertion start and stop positions are 107 and 116, respectively (spliced positions are shown in bold).

Heavy chain sequence:

EVQLVESGGGLVQPGGSLRLSCAASGFNIKDTYIHWVRQAPGKGLEWVARIYPTNGYTRYADSVKG

RFTISADTSKNTAYLQMNSLRAEDTAVYYCSRWGGDGFYAMDYWGQGTLVTVSSA

Light chain sequence:

DIQMTQSPSSLSASVGDRVTITCRASQDVNTAVAWYQQKPGKAPKLLIYSASFLYSGVPSRFSGSR

SGTDFTLTISSLQPEDFATYYCQQHYTTPPTFGQGTKVEIKR

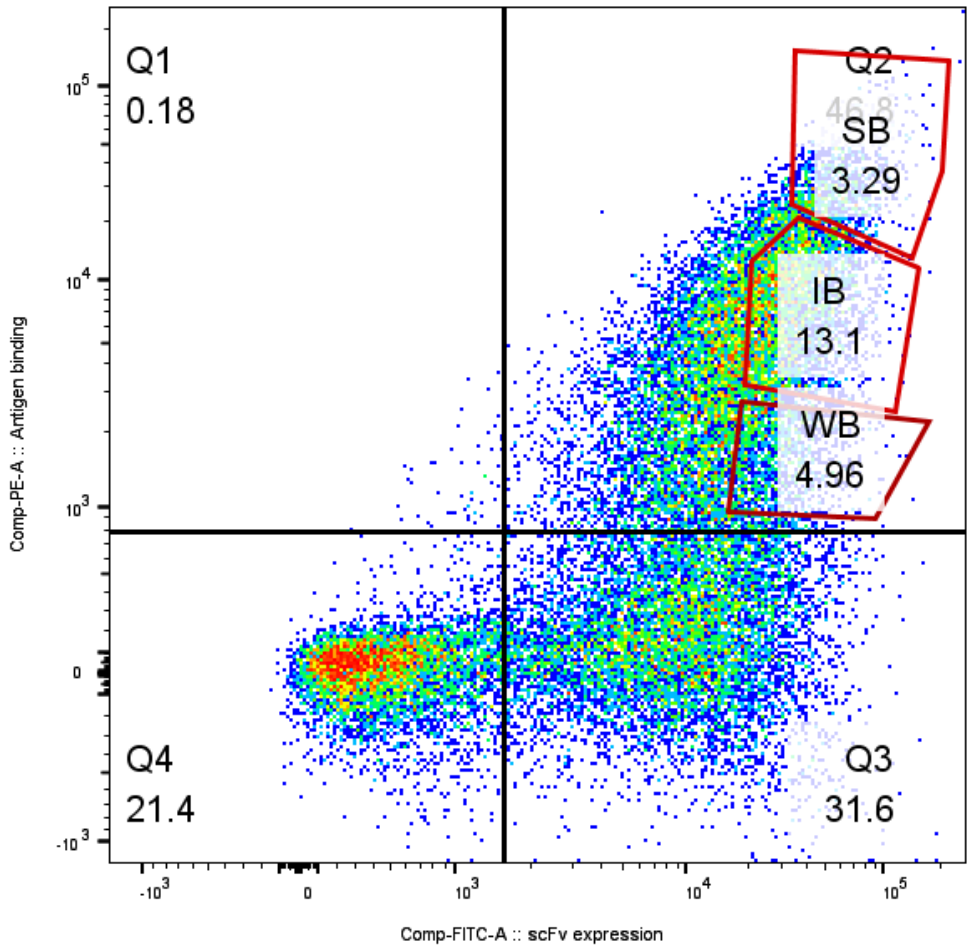

Figure 6.0.1: Bivariate flow-cytometric analysis of our Trastuzumab-variant library highlights different antigen-binding populations. Cells are double-labelled with biotinylated antigen/streptavidin–phycoerythrin (y-axis), and anti-V5/anti-mouse FITC labels (x-axis). The percentages of events measured in each quadrant and covered by each gate are displayed as numbers. The sub-population with the brightest antigen labelling at a given scFv expression is termed 'strong binder' (SB), referred to in Chapter 3 as 'high-affinity'. Sub-populations showing intermediate (medium-affinity) and weak (low-affinity) labelling for HER2 at a given scFv expression are termed intermediate (IB) and weak binders (WB).

Classification methods outperform clonotyping

To baseline our ML classification methods, we used clonotyping with our HER2-aff-large dataset. ANARCI was first used to assign V and J-genes and sequences were then clustered by these annotations and 70% CDRH3 sequence identity. This process resulted in 14,786 clusters, the largest of which contained 11,623 members while the smallest clusters were singletons. As expected, all sequences shared the same V-gene, 'IGHV3-66' (V-gene influence stops at IMGT position 106). J-gene usage, which affects IMGT positions 113 onwards, was split between 'IGHJ4' (14,236 clusters) and 'IGHJ1' (550 clusters).

Test set sequence labels were then 'predicted' based on whether or not they belonged to the same cluster as positive (binding) sequences from the train or validation datasets. If a test set sequence belonged to a cluster containing one or more positive sequences from the train or validation datasets then that test set sequence was predicted to be positive too; otherwise, the sequence was predicted to be non-binding.

We found that both FLAML and the CNN outperformed clonotyping across all data splits we examined (see Figure 6.0.2). The performance of clonotyping plateaued/fell as the train and validation dataset size increased beyond ∼5k sequences as most test sequences were eventually found to belong to a cluster containing a binding sequence from the train or validation data sets. Note - clonotyping's ROC AUC is expected to fall to 0.5 once all test data is predicted to belong to the positive class, while the PR AUC will approach 0.66, equal to $1-\frac{1}{2}(1-imbalance)$.

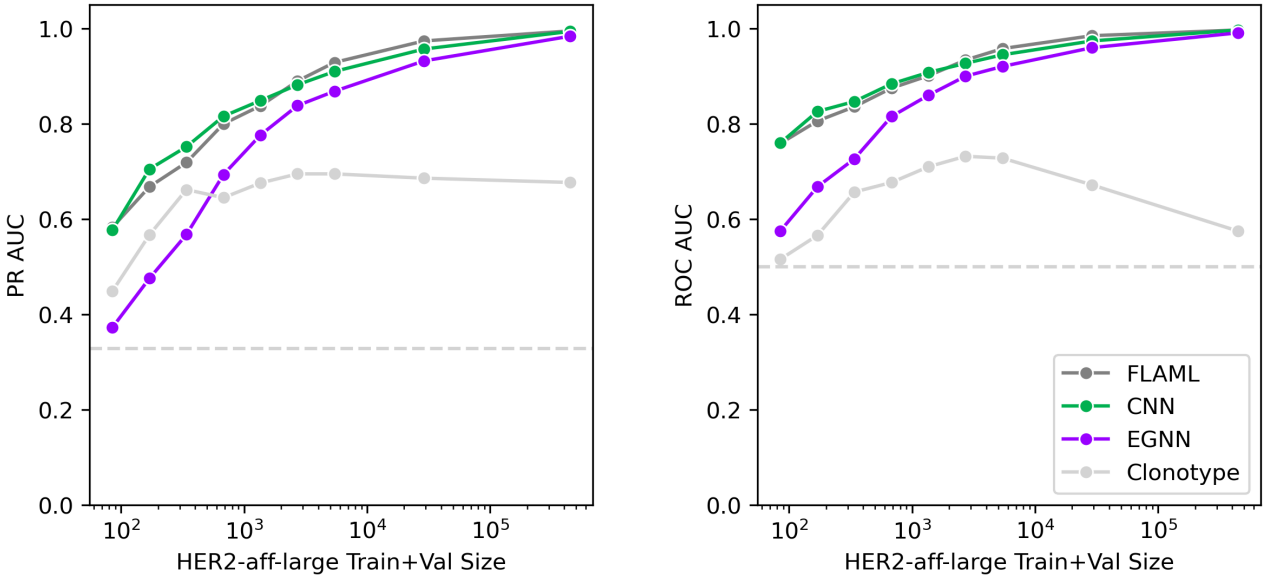

Figure 6.0.2: Areas under the Precision-Recall (PR AUC) and Receiver Operating Characteristic curves (ROC AUC) for all binder classification methods (FLAML, CNN, and EGNN). Results are shown on our HER2-aff-large dataset with overlapping sequences removed. Train, validation, and test sets are split randomly. Train and validation sets have a relative size ratio of 70:15. All data not assigned to the train or validation dataset is used as the test set, on which the results are presented. Random guessing would result in a PR AUC equal to HER2-aff-large's class imbalance of 0.33 and a ROC AUC of 0.5 (light grey dashed lines). Clonotyping (light grey solid line) offers PR AUC and ROC AUC performances above random guessing but below both FLAML and the CNN at all data splits.

CNN classification results on HER2-aff-large

To allow easier comparisons to future methods, we provide a detailed breakdown of our CNN's performance on different training dataset sizes (see Table 6.0.1). All training and test datasets can be found at doi.org/10.5281/zenodo.10549114.

Train size	PR AUC	F1	MCC	ROC AUC	BA
85	0.577	0.605	0.373	0.760	0.698
170	0.705	0.666	0.480	0.826	0.752
340	0.752	0.685	0.514	0.847	0.768
680	0.816	0.729	0.585	0.884	0.803
1,360	0.849	0.764	0.640	0.908	0.830
2,720	0.882	0.794	0.687	0.927	0.853
5,440	0.910	0.825	0.735	0.945	0.876
28,941	0.957	0.883	0.824	0.974	0.919
445,694	0.994	0.966	0.949	0.997	0.976
85	0.422	0.520	0.199	0.639	0.605
170	0.415	0.543	0.234	0.643	0.619
340	0.529	0.580	0.327	0.732	0.674
680	0.522	0.548	0.272	0.703	0.645
1,360	0.619	0.602	0.367	0.770	0.695
2,720	0.703	0.665	0.476	0.831	0.752
5,440	0.757	0.693	0.524	0.858	0.775
28,941	0.848	0.751	0.620	0.905	0.819
445,694	0.994	0.967	0.950	0.997	0.977

Table 6.0.1: Performance of our CNN trained on varying amounts of HER-aff-large data. Each 'Train size' stated in the table comprises both train and validation datasets using a 70-15 split. Performances are calculated on the corresponding test sets - all HER2-aff-large data not included in the train or validation dataset. For each train set, the CNN is trained with a single random seed and some variation can be expected with repeated training runs. The top half of the table shows the CNN performance when the HER2-aff-large dataset is randomly split, and the lower half of the table shows the results when the data is split by clonotype. PR AUC = Area Under the Precision-Recall Curve; F1 = F1-score, MCC = Matthews Correlation Coefficient; ROC AUC = Area Under the Receiver Operating Characteristic Curve; BA = Balanced Accuracy.

CNN classification results for an anti-influenza dataset

We apply our CNN, without further optimisation, to classify anti-influenza antibodies (CR9114 variants) for binding against three strains of the influenza surface protein hemagglutinin (HA). We use data collected by Phillip *et al.* (Phillips et al. 2021) and compare to results from Bachas *et al.* (Bachas et al. 2022)

The binding affinity datasets of CR9114 variants measured against the three HA strains - H1, H3, and FluB - contain very different class imbalances of 97%, 11%, and 0.3%, respectively. We maintain this class imbalance for the training, validation, and test datasets as closely as possible. All training and test datasets can be found at doi.org/10.5281/zenodo.10549114.

We select the best results from Bachas *et al.* for comparison against our CNN. The CNN achieves higher Balanced Accuracies for most strains and data splits (see Table 6.0.2). However, the CNN performance is lower than Bachas *et al.* for small training set sizes (65), often when the train and/or validation dataset comprises examples of only one class (binding or non-binding) due to the severe class imbalances involved. During training, we could have artificially corrected this class imbalance and intentionally over-sampled the minority class, but this is likely not possible in real research settings, so we left the results as they stand.

Target	Train size	BA (ours)	BA (Bachas *et al.*)
	6,509	**0.964**	0.92
H1	651	**0.906**	0.90
	65	0.712	**0.73**
	6,509	**0.987**	0.98
H3	651	**0.961**	0.95
	65	0.799	**0.91**
	6,509	**0.992**	0.96
FluB	651	**0.742**	0.64
	65	0.500	**0.51**

Table 6.0.2: Performance of our CNN compared to Bachas *et al.* when trained on varying amounts of CR9114 binding data. The 'Train size' stated in the table comprises both train and validation datasets using a 50-15 split. The exact training sets used by Bachas *et al.* are not shared and may differ from ours. Our performance on the smallest train and validation dataset sizes is not optimal as the validation dataset (and sometimes the train set) often consists of none of one class given the severe class imbalances involved. The best-performing method for each strain and train-set size is shown in bold. BA = Balanced Accuracy.

Obtaining likelihoods from BLOSUM matrices

BLOSUM matrices (Henikoff et al. 1992) (BLOcks SUbstitution Matrices) describe which amino acid substitutions are most likely to be observed in nature. These matrices are 20×20 integer arrays where positive and negative values indicate likely and unlikely substitutions respectively. To obtain likelihoods from these matrices for library design, we reverse-engineered these matrices with antibody CDRH3 loops in mind.

First, we selected the BLOSUM-45 matrix which is more suitable for distantly related proteins as our starting point. We chose BLOSUM-45, rather than the more commonly used BLOSUM-62 matrix, as CDRH3 loops are dominated by Somatic Hypermutations (SHMs) rather than the gradual evolution observed in other proteins.

Next, we followed the calculations covered by Eddy (Eddy 2004) to reverse engineer the BLOSUM-45 matrix scores, $s(a, b)$, to obtain the target frequencies, p_{ab}

$$s(a, b) = \frac{1}{\lambda} log \frac{p_{ab}}{f_a f_b}$$

The background frequencies, $f_{a,b}$, were obtained by counting how often each of the 20 standard amino acids appeared in the CDRH3 loops of structures from SAbDab (Dunbar et al. 2014; Schneider et al. 2022) (counts obtained 09/02/23).

After visual inspection of the resulting logo plots, we opted to fix lambda to be 0.25, rather than the classically used $\frac{1}{2} log \sqrt{2} \sim 0.35$. This choice was made to reduce how often the original amino acid would be selected, increasing the diversity of our subsequently designed library. Finally, the probabilities, p_{ab}, were normalised to sum to one. The code to perform this full calculation can be found at github.com/oxpig/Tz_her2_affinity_and_beyond.

BLOSUM, AbLang, ESM, and ProteinMPNN's predicted binders cover diverse areas of sequence space

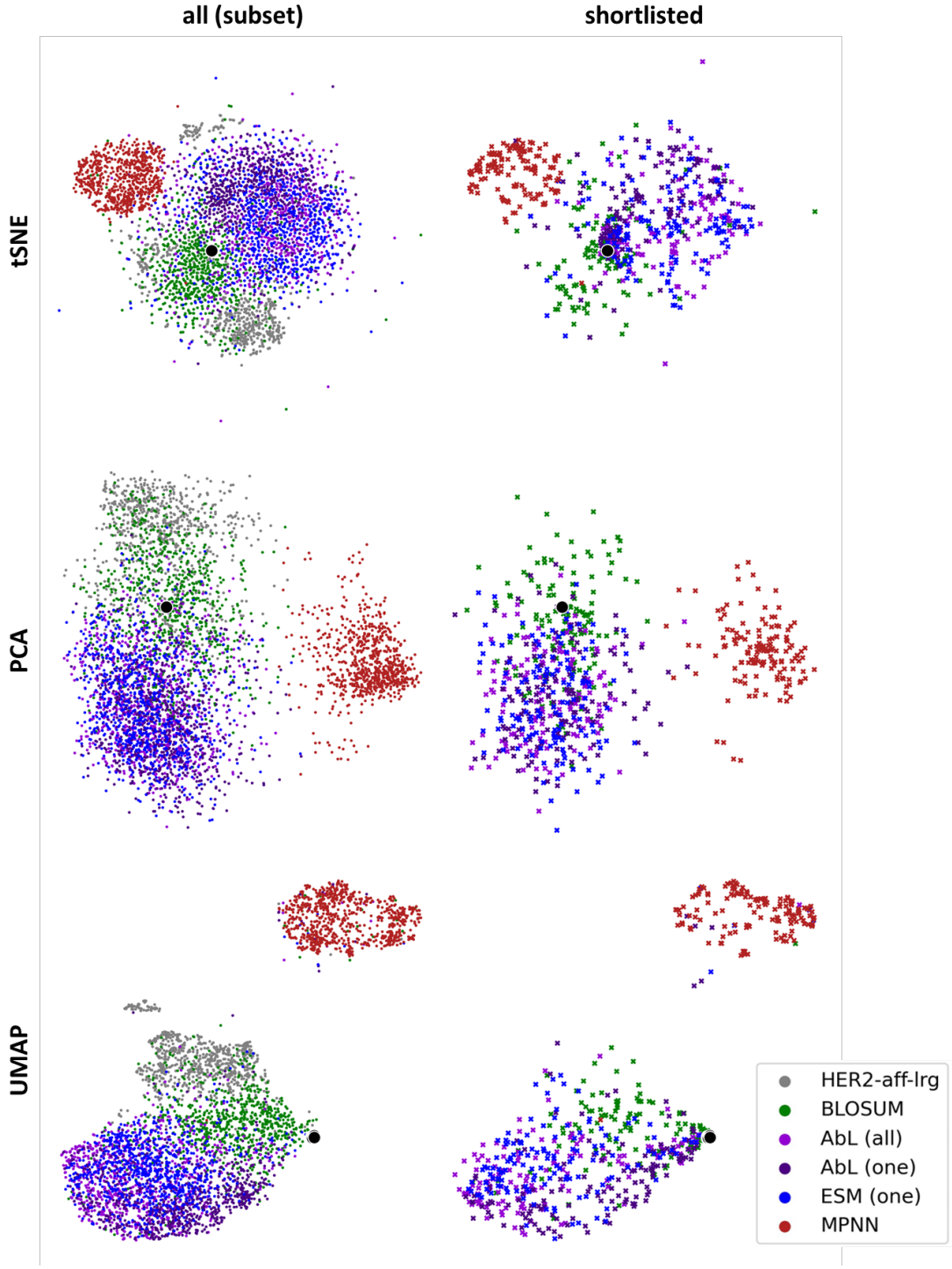

Figure 6.0.3: all (subset): t-SNE, PCA, and UMAP plots comparing the sequence space explored by different methods when designing HER2 binders. ∼800 sequences designed by each method are shown and compared to HER2-aff-large. Each 2D visualisation takes as input the flattened one-hot encodings of the designed CDRH3 loops (residues 107 to 116). **shortlisted**: corresponding 2D visualisations of sequences shortlisted for experimental validation with CNN-HER2-max binding probabilities above 90%.

Edit distance distributions of designed libraries

For each library method, ∼1k sequences were sub-sampled from a maximum of 1m generated sequences to match the edit distance distribution observed in HER2-aff-large, where possible. This sub-sampling is important when comparing computationally designed libraries to experimental data as smaller edit distances contain larger proportions of high-affinity variants.

Due to the nature of the sampling distributions (see Figure 3.3.2), some methods failed to efficiently sample small edit distances (see Figure 6.0.4). The longer run-time and constrained distribution of sequences created by ProteinMPNN in particular meant it was not possible to sub-sample from these to match the edit distance distribution of HER2-aff-large. Instead, 1k sequences were randomly sampled from all non-redundant ProteinMPNN designs for comparison against other methods.

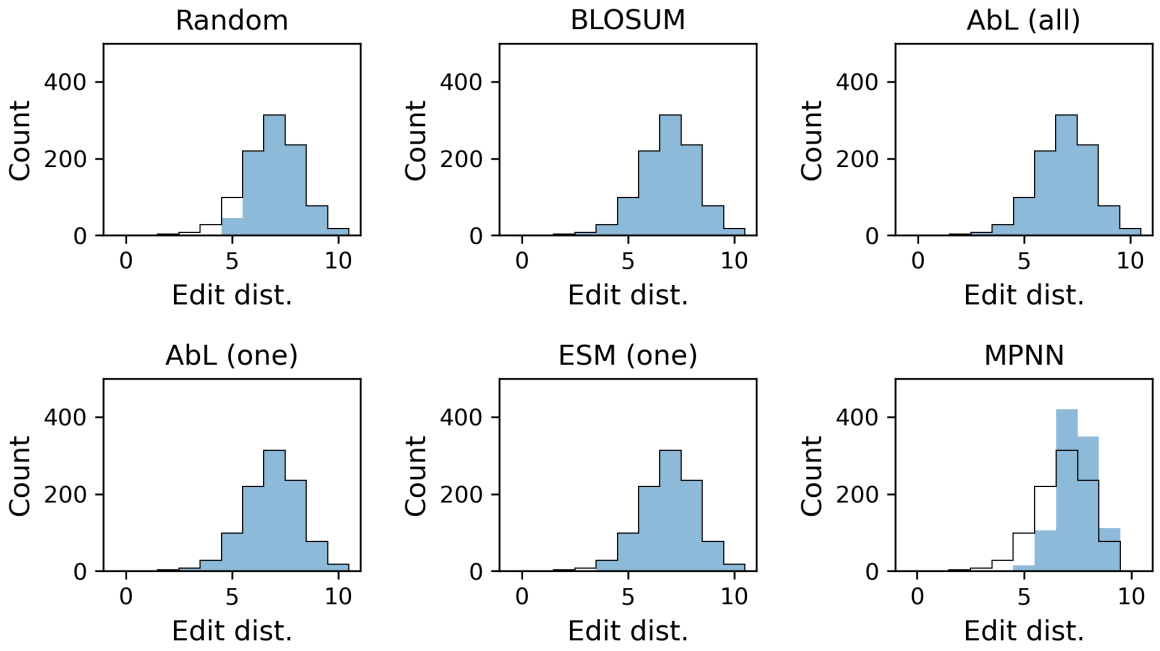

Figure 6.0.4: Comparison of the edit distributions of the ∼1k member libraries designed using each method (solid blue) vs the underlying distribution observed in HER2-aff-large (black line). HER2-aff-large's edit distance distribution peaked at an edit distance of seven, and few sequences were observed with very small or very large edit distances from Trastuzumab. BLOSUM, AbLang, and ESM largely succeeded in generating enough sequences at each edit distance given 1m attempts. Random mutations however did not generate many sequences with small edit distances. Large edit distance designs dominated ProteinMPNN's output so it was not possible to sub-sample from these to match the edit distance distribution of HER2-aff-large; instead, 1k sequences were randomly selected from all its designs.

BLOSUM, AbLang, ESM, and ProteinMPNN generate antibody libraries with high proportions of predicted binders, even when restricted to large edit distance designs

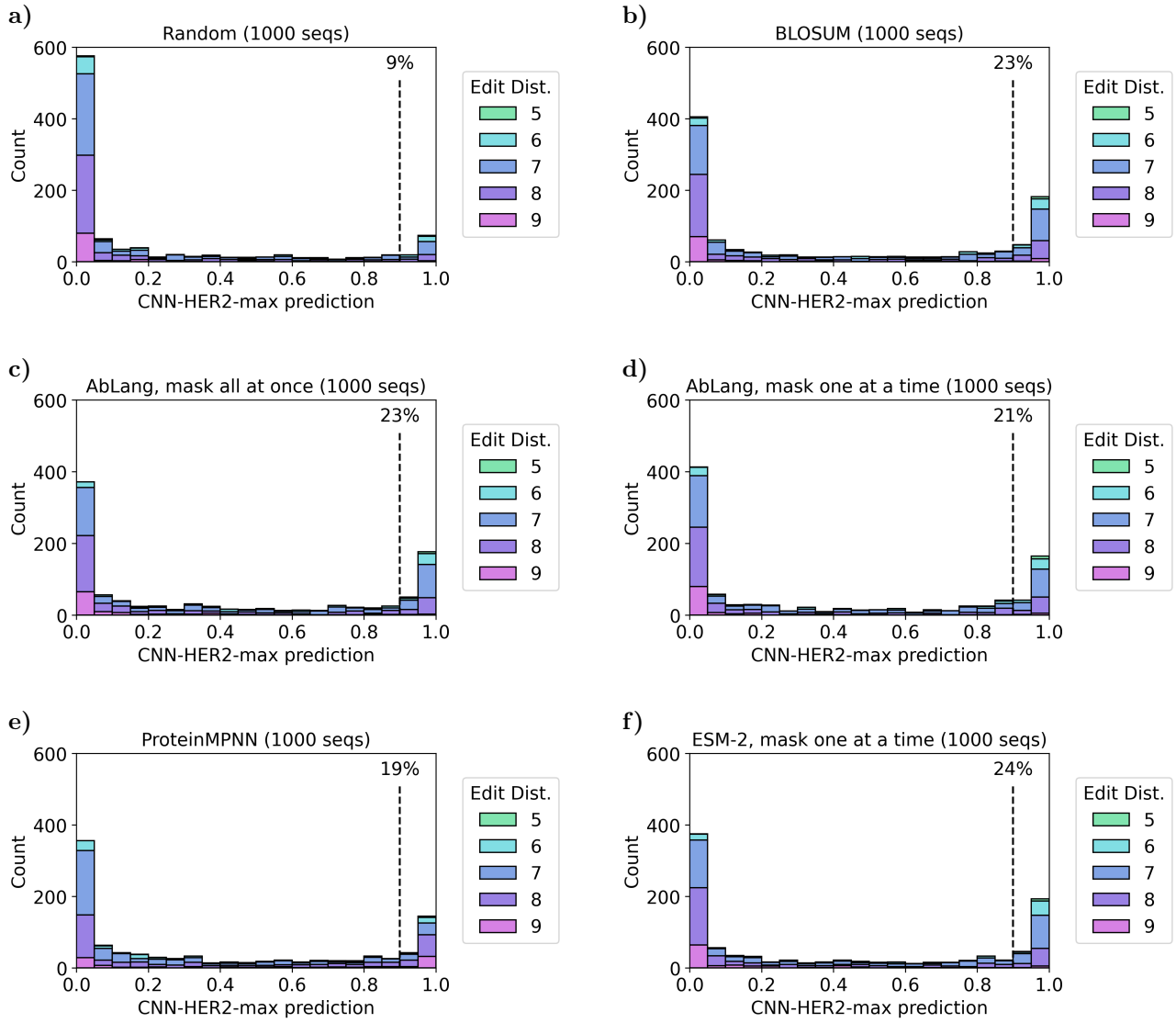

Figure 6.0.5: Distributions of HER2 binding predictions for 1k sequences generated using BLOSUM, AbLang, ESM, and ProteinMPNN, plus a random baseline. These sequences were sub-sampled from a maximum of 1m generated sequences to match the edit distance distribution observed in ProteinMPNN's designs (Figure 6.0.4). The edit distance distribution of ProteinMPNN's designs is shifted towards larger edit distances compared to HER2-aff-large. This shift explains the reduction in predicted binding enrichments for all methods (excluding ProteinMPNN) compared to Chapter 3.

ESM-2 designed libraries - masking all CDRH3 residues at once vs one at a time

Predicted enrichments of ESM-2 designed libraries varied considerably when masking the whole CDRH3 at once compared to masking residues one at a time (see Figure 6.0.6). Masking one residue at a time offered higher predicted enrichments, so only designs from this library were selected for experimental validation using BLI. Note that a single random seed was used to generate each library and some variation can be expected with repeated rounds of design.

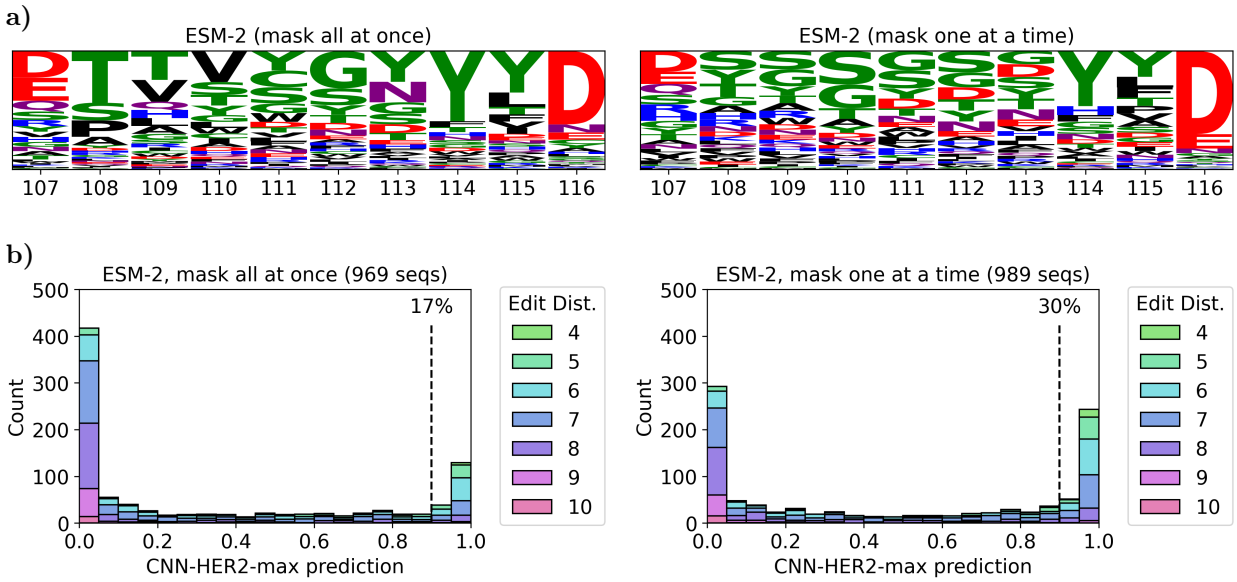

Figure 6.0.6: Comparison of ESM-2-designed Trastuzumab-variant libraries when masking the entire CDRH3 loop at once vs masking one residue at a time. **a)** shows the logo plots of the raw weighted sampling distributions resulting from each approach. **b)** shows the distribution of HER2 binding predictions for ∼1k sequences generated from each distribution. As before, these sequences were sub-sampled from a maximum of 1m generated sequences to match the edit distance distribution observed in HER2-aff-large, where possible. Masking CDRH3 residues one at a time more efficiently explored smaller edit distances from Trastuzumab and resulted in higher predicted enrichments compared to masking all CDRH3 residues at once.

Edit distance distribution of predicted binders to CDRH3s in OAS

KASearch (Olsen et al. 2023) was used to calculate the edit distances of our predicted binding designs to the closest sequences from OAS (Kovaltsuk et al. 2018; Olsen et al. 2022b). The search was performed using the entire aligned OAS database of 2.4 billion sequences (11/01/2023). We limited the search to human CDRH3s and searched for matches between IMGT positions 107 to 116 only.

The same calculation was made for Shanehsazzadeh *et al.*'s 198 length-matched SPR-experimentally confirmed binders. We observed that Shanehsazzadeh *et al.*'s and AbLang's designs tended to most closely resemble previously observed CDRH3 sequences (see Figure 6.0.7). BLOSUM's and ProteinMPNN's designs were the most novel, with most edit distances of two or three from their closest sequences in OAS.

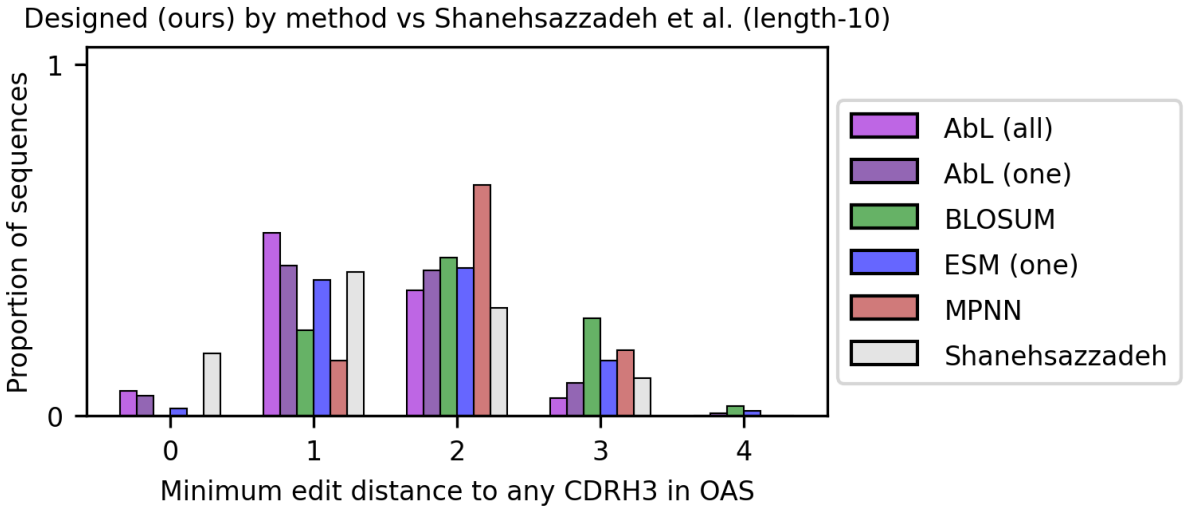

Figure 6.0.7: Comparison of how similar each library design method's predicted binders are to their closest sequences in OAS. Shanehsazzadeh *et al.*'s 198 length-matched SPR-experimentally confirmed binders are also shown in light grey. Edit distances to OAS were calculated using KASearch and compared to human CDRH3 loops only. If a designed CDRH3 exists in OAS it will have an edit distance of zero. More novel designs, further from any previously observed CDRH3, will have larger edit distances.

Binding predictions of experimentally shortlisted sequences by CNN trained on different data

BLI shall be performed on 700 Trastuzumab-variants with CNN-HER2-max predictions above 90% (see Figure 6.0.8, left). This cut-off was made to increase our odds of testing binding sequences.

We also checked if these 700 sequences were predicted to bind using the same CNN trained on different datasets. First, the CNN was retrained on data from Mason *et al.*, removing sequences that belonged to both positive and negative classes and using a 70-15-15 train-validation-test split. The predicted enrichments of the shortlisted sequences dropped sharply using this retrained model (see Figure 6.0.8, right).

A new, small dataset was also constructed using all 198 length-ten binding sequences from Shanehsazzadeh *et al.* and a random subset of 376 negative variants from HER2-aff-large. When retrained on this new dataset, the CNN offered high binding probabilities for most of the shortlisted sequences, though ProteinMPNN's designs were still ranked low (see Figure 6.0.8, bottom).

As consensus was rare between all the differently trained CNNs, we continued to use only the predictions of the CNN trained on HER2-aff-large when shortlisting sequences for testing due to the larger training set size.

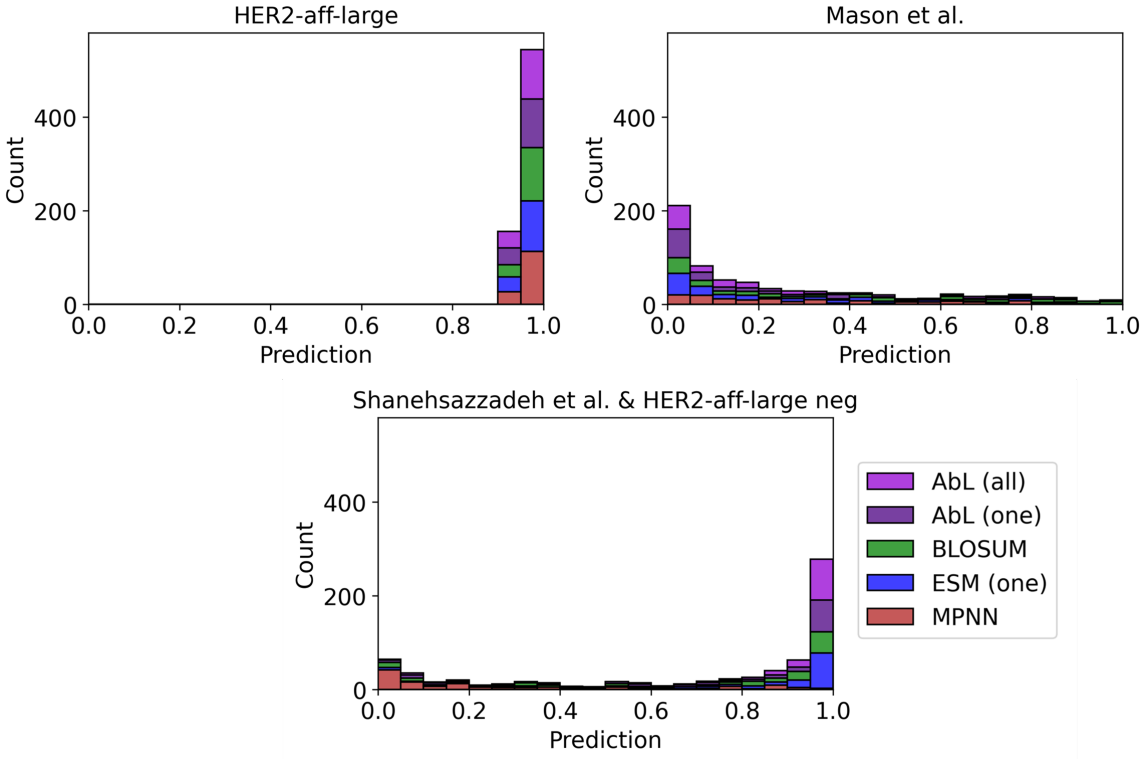

Figure 6.0.8: Binding predictions from three CNNs trained on different data sources for our 700 novel Trastuzumab variants shortlisted for experimental validation. Variants were shortlisted based on having CNN-HER2-max binding probabilities above 90% (top left). When the same CNN was retrained on smaller datasets from Mason *et al.* (top right) and Shanehsazzadeh *et al.* (bottom) the number of predicted binders within our shortlisted sequences fell.

7 | Appendix B for Chapter 4

Humatch classification accuracy when trained and tested on Hu-mAb's data

In Chapter 4, we highlight how Humatch was trained on a more extensive dataset than used by Hu-mAb (Marks et al. 2021), and that training was stopped early to facilitate smoother humanisation. Table 7.0.1 demonstrates that Humatch's architecture is capable of achieving comparably high accuracies to Hu-mAb when trained and tested on the same dataset. Here, Humatch's CNN-H and CNN-L were trained for three epochs, similarly to Chapter 4, but this time on Hu-mAb's data. The data consists of only 152 IMGT residue positions, the light chain data lacks V-genes LV9 and KV7, and neither heavy nor light datasets contain rhesus sequences in the negative training data. Table 7.0.1 shows Humatch achieved perfect PR AUC and ROC AUC scores for all V-genes measured to three decimal places.

Class	PR AUC	F1	MCC	ROC AUC	BA
HV1	1.000	1.000	1.000	1.000	1.000
HV2	1.000	1.000	1.000	1.000	1.000
HV3	1.000	1.000	1.000	1.000	1.000
HV4	1.000	1.000	1.000	1.000	1.000
HV5	1.000	1.000	1.000	1.000	1.000
HV6	1.000	1.000	1.000	1.000	1.000
HV7	1.000	1.000	1.000	1.000	1.000
LV1	1.000	1.000	1.000	1.000	1.000
LV2	1.000	1.000	1.000	1.000	1.000
LV3	1.000	1.000	1.000	1.000	1.000
LV4	1.000	1.000	1.000	1.000	1.000
LV5	1.000	1.000	1.000	1.000	1.000
LV6	1.000	1.000	1.000	1.000	1.000
LV7	1.000	1.000	1.000	1.000	1.000
LV8	1.000	1.000	1.000	1.000	1.000
LV10	1.000	0.998	0.998	1.000	1.000
KV1	1.000	1.000	1.000	1.000	1.000
KV2	1.000	1.000	1.000	1.000	1.000
KV3	1.000	1.000	1.000	1.000	1.000
KV4	1.000	1.000	1.000	1.000	1.000
KV5	1.000	1.000	1.000	1.000	1.000
KV6	1.000	1.000	1.000	1.000	1.000

Table 7.0.1: Performance of Humatch's heavy and light CNN classifiers trained and tested on HumAbs dataset. Sequences belonging to all genes are classified with near-perfect accuracy. PR AUC = Area Under the Precision-Recall Curve; F1 = F1-score, MCC = Matthews Correlation Coefficient; ROC AUC = Area Under the Receiver Operating Characteristic Curve; BA = Balanced Accuracy.

Heavy-light gene pairings of Humatch's true and artificially paired training data

The artificially paired data used to train Humatch must have approximately the same heavy-light gene pairings as the true data so that CNN-P cannot simply learn to identify unusual pairings. Figure 7.0.1 shows that both datasets have similar distributions of pairings, with most pairs belonging to heavy V-genes 1, 3, and 4. Most light genes belong to Kappa genes 1-4 and Lambda genes 1-3.

Figure 7.0.1: Distribution of heavy-light gene pairings for the true (top) and artificially paired (bottom) datasets used to train Humatch. The total counts for each gene pair are given in the respective cells and these are coloured by their proportion of the total dataset (low-high, purple-yellow).

Humatch anti-drug antibody correlation

Figure 7.0.2 shows how each CNN contributes to screening for therapeutics that may illicit anti-drug antibody (ADA) responses in high proportions of patients. All CNNs offer some separation of low- and high-ADA antibodies, though CNN-H and CNN-L both include one high-ADA therapeutic in their top-ranked bin.

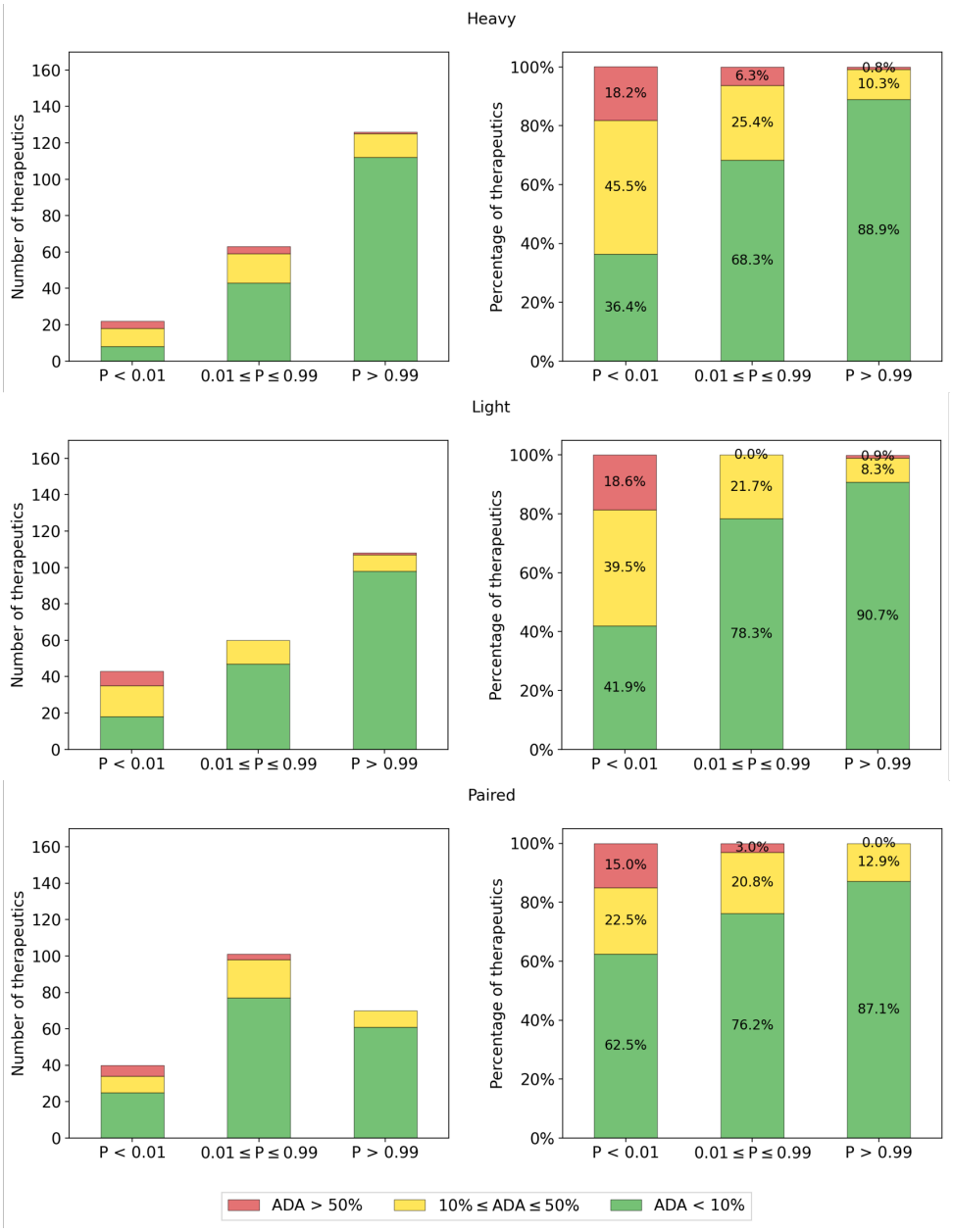

Figure 7.0.2: Comparison of how each CNN separates 211 therapeutics with different anti-drug antibody (ADA) levels.

In Chapter 4, we highlight how the minimum of the three scores is recommended to remove most high ADA therapeutics. However, Figure 7.0.3 shows that the maximum and mean of the three scores are also efficient at screening out only the most immunogenic therapeutics while maintaining most others.

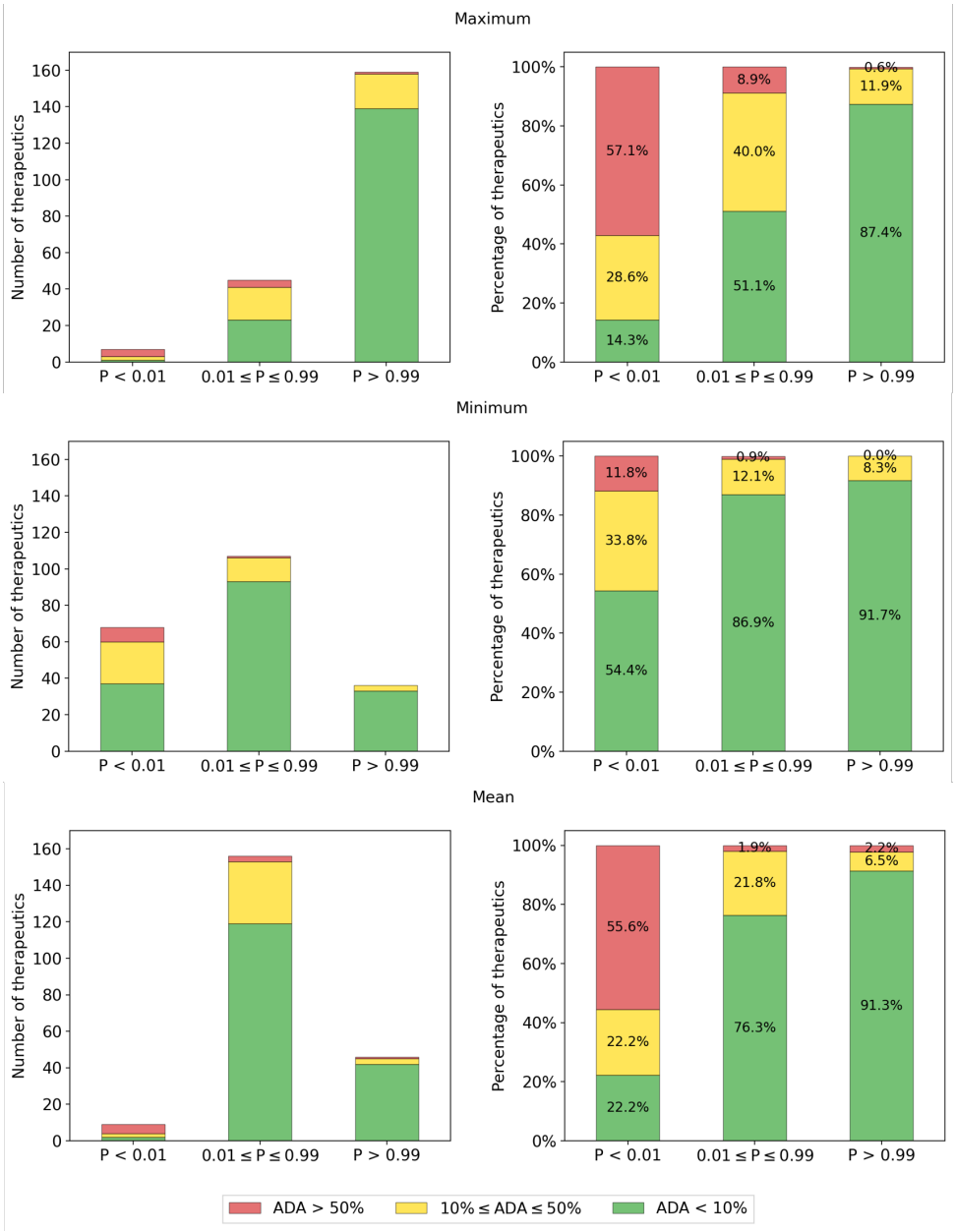

Figure 7.0.3: Comparison of how the maximum, minimum, or mean of the three CNNs separates 211 therapeutics with different anti-drug antibody (ADA) levels.

Hu-mAb, AbNatiV, and OASis anti-drug antibody correlation

Hu-mAb (Marks et al. 2021), AbNatiV (Ramon et al. 2024), and OASis (Prihoda et al. 2022) were used to score the same 211 therapeutics with ADA information as Humatch. These tools were run with default and paper-recommended parameters.

Similarly to Humatch, the minimum of each tool's heavy and light chain scores were taken and used to group the 211 therapeutics into three bins (see Figure 7.0.4). The bin thresholds were selected to yield approximately the same number of therapeutics in each bin as Humatch.

We found that all methods placed the same number of high ADA (red) therapeutics in the bottom bin. Hu-mAb placed proportionally more medium (yellow) ADA therapeutics in this bin than other methods. Hu-mAb and OASis also achieved cleaner top bins than Humatch and AbNatiV, with all (or all but one) therapeutics having low (green) ADA levels.

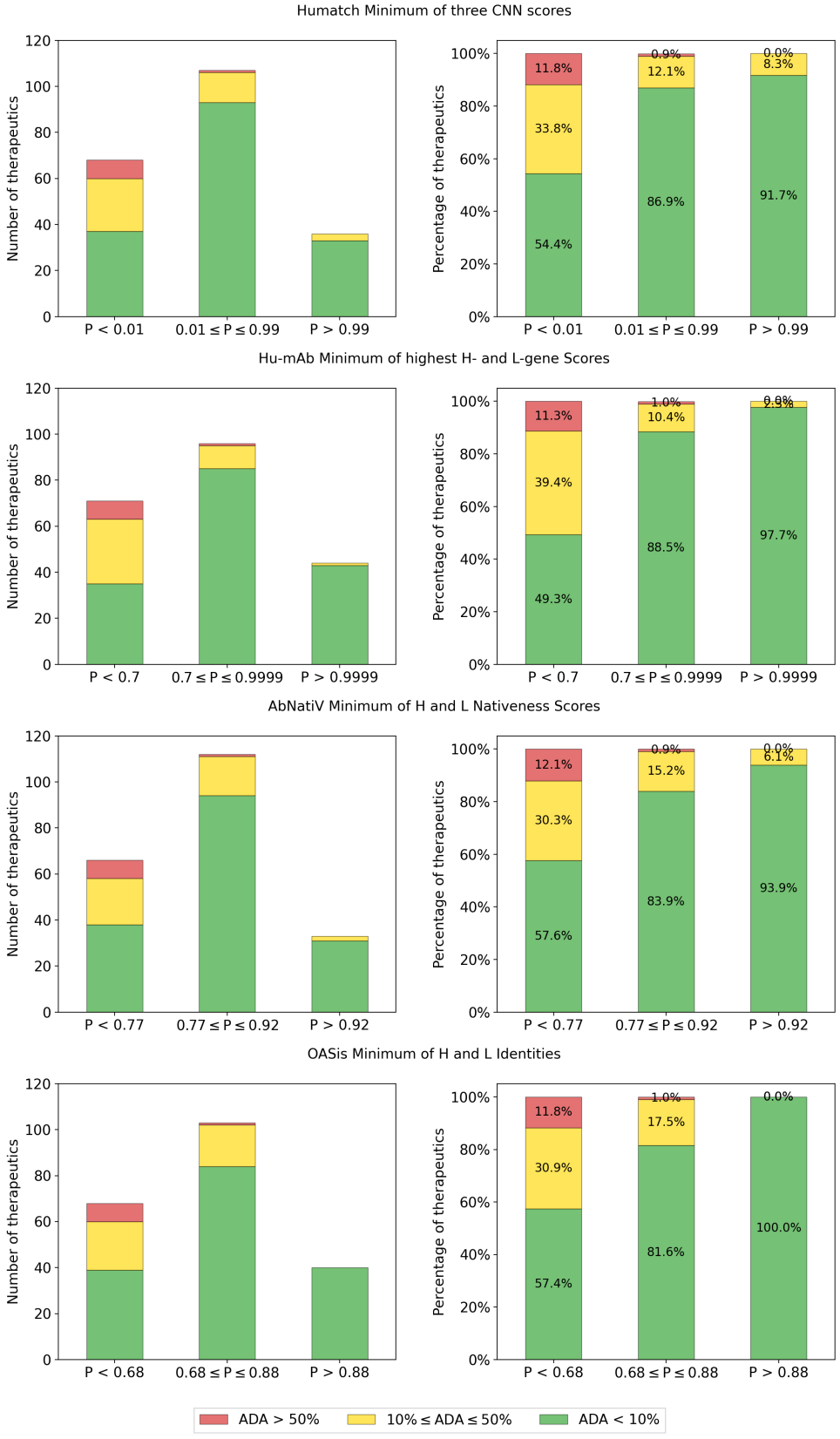

Figure 7.0.4: Comparison of how Humatch, Hu-mAb, AbNatiV, and OASis separate 211 therapeutics with different anti-drug antibody (ADA) levels.

Choosing optimum CNN classifier cutoffs

The data used for training Humatch suffers from large gene-class imbalances. Therefore, to determine optimum classifier cut-offs, we used sample weights to counter this imbalance. We recommend using a threshold of 0.95 for all classifiers as this was found to provide high sample-weighted precision and recall for all classes (see Table 7.0.2). We prioritised near-perfect precision over recall in selecting this threshold given the nature of the humanisation problem.

Class	Precision	Recall
HV1	0.999	0.946
HV2	1.000	0.980
HV3	0.999	0.945
HV4	1.000	0.959
HV5	1.000	0.975
HV6	1.000	0.780
HV7	1.000	0.913
LV1	1.000	0.998
LV2	1.000	0.994
LV3	1.000	0.997
LV4	1.000	1.000
LV5	1.000	0.980
LV6	1.000	1.000
LV7	1.000	0.981
LV8	1.000	1.000
LV9	1.000	0.995
LV10	1.000	0.969
KV1	1.000	0.973
KV2	1.000	0.988
KV3	1.000	0.989
KV4	1.000	0.997
KV5	1.000	0.998
KV6	1.000	0.993
KV7	1.000	0.295
true pairs	0.995	0.933

Table 7.0.2: Comparison of the sample-weighted precision and recall achieved by each class using classifier cut-offs of 0.95. The sample-weighting used equalised the class imbalance between the target class and all other classes in each instance. The precision exceeds 0.99 for all CNN classes. The recall was high for most classes, though dropped when little training data was available e.g. KV7.

Humatch classification accuracy remains high when data is split by allele

To test whether Humatch's predictions might be robust to classifying new human alleles yet to be recorded, we split the training data so that the test set contained only alleles unseen during training for heavy V-genes 1-4. We selected only heavy V-genes 1-4 for this task as others did not allow us to obtain sufficient data to maintain the same train-validation-test dataset sizes and proportions as used in Humatch's full training.

CNN-H was therefore retrained with the same negative data and heavy V-genes 5-7 as Chapter 4. HV1 alleles 1-11, HV2 alleles 1-4, HV3 alleles 1-4, and HV4 alleles 1-7 were also used for training and validation. Other HV1-4 alleles were reserved for testing, alongside the same negative and HV5-7 data as before. The training was stopped after three epochs, similar to Chapter 4.

Tables 7.0.3 & 7.0.4 show that Humatch's test set performance remains high for all genes, regardless of whether or not the data was split by allele. The largest drop in performance was for HV2 which may be because the amount of training data available is an order of magnitude lower than HV1, 3, and 4.

Class	PR AUC	F1	MCC	ROC AUC	BA
HV1	1.000	0.998	0.998	1.000	1.000
HV2	0.949	0.909	0.911	0.998	0.918
HV3	0.995	0.952	0.936	0.998	0.975
HV4	1.000	0.990	0.987	1.000	0.996
HV5	1.000	0.987	0.987	1.000	1.000
HV6	0.991	0.811	0.825	1.000	0.999
HV7	0.991	0.914	0.915	1.000	0.987

Table 7.0.3: Performance of Humatch's heavy CNN classifier, CNN-H, when trained and tested on data split by allele (HV1-4 only). Sequences belonging to all genes are classified with high accuracy. Gene classes with the lowest scores, such as HV2, were tested on unseen alleles and have comparatively little training data available. PR AUC = Area Under the Precision-Recall Curve; F1 = F1-score, MCC = Matthews Correlation Coefficient; ROC AUC = Area Under the Receiver Operating Characteristic Curve; BA = Balanced Accuracy.

Class	Precision	Recall
HV1	0.999	1.000
HV2	1.000	0.820
HV3	0.998	0.888
HV4	0.999	0.986
HV5	1.000	0.997
HV6	1.000	0.909
HV7	1.000	0.913

Table 7.0.4: Comparison of the sample-weighted precision and recall achieved by each gene using classifier cut-offs of 0.95 for CNN-H trained on allele-split data. The sample-weighting used equalised the class imbalance between the target class and all other classes in each instance. The precision exceeds 0.99 for all CNN classes. The recall was high for most classes, though dropped when little training data was available and the test set included previously unseen alleles e.g. HV2.

Baseline humanisation of 25 therapeutics using top germline mutations only

We conducted a baseline experiment where only the top germline mutations were made to test the importance of Humatch's CNN guidance in the humanisation process. In this baseline, the same number of heavy and light chain mutations were made as in Humatch's full humanisation logic. We found that no baseline-designed paired sequences matched Humatch's designs exactly and only two designs passed all three CNN target thresholds of 0.95 (see Table 7.0.5). This baseline shows that Humatch considers the sequence context when deciding which germline mutations should be made.

Therapeutic	H-edit	L-edit	CNN-H	CNN-L	CNN-P
AntiCD28	8	10	1.00	1.00	0.00
Bevacizumab	6	6	0.71	0.00	1.00
Campath	10	4	0.00	1.00	1.00
Certolizumab	4	2	1.00	0.37	1.00
Clazakizumab	2	0	1.00	0.99	1.00
Crizanlizumab	6	4	0.99	0.05	0.00
Eculizumab	8	6	1.00	0.04	0.63
Etaracizumab	10	4	1.00	1.00	0.03
Herceptin	4	2	0.97	0.99	1.00
Idarucizumab	2	4	0.92	0.10	1.00
Ixekizumab	4	9	0.99	0.70	1.00
Ligelizumab	4	2	0.55	1.00	0.97
Lorvotuzumab	6	11	1.00	0.83	0.00
Mogamulizumab	4	6	0.53	0.03	0.99
Omalizumab	6	5	0.95	0.02	0.08
Palivizumab	10	8	0.00	1.00	0.25
Pembrolizumab	8	2	0.99	1.00	0.32
Pertuzumab	4	2	0.47	0.86	0.99
Pinatuzumab	4	6	1.00	0.10	0.98
Refanezumab	10	4	0.98	0.48	0.99
Reslizumab	4	4	1.00	0.01	0.97
Rovalpituzumab	5	2	0.99	1.00	0.55
Solanezumab	4	8	0.00	0.03	0.94
Talacotuzumab	10	8	0.46	0.73	1.00
Tocilizumab	4	6	0.12	0.00	1.00

Figure 7.0.5: CNN scores achieved by baseline germline-only humanisation method. CNN thresholds of 0.95 passed are shown in green. Lower scores are shown in red. Only 2/25 baseline designs passed all three thresholds.

Humatch humanisation of 25 precursor therapeutics using a lower initial germline-likeness

The initial germline-likeness (GL) targeted by Humatch can be adjusted by the user. By default, Humatch uses a target of 0.40, corresponding to the lower 20th percentile of GL scores achieved by clinical-stage therapeutics (CSTs). Lowering this target can allow Humatch to design sequences that pass all CNN thresholds of 0.95 in fewer edits than otherwise possible at the expense of lower alignment with experimental endpoints. Table 7.0.5 shows Humatch's performance when humanising 25 precursor therapeutics using GL targets of 0.40 vs 0.38 (approximately corresponding to the lower 10th percentile of GL scores achieved by CSTs, see Figure 4.3.2). Fewer edits are made to both heavy and light chains using the lower GL score, but the overlap of Humatch's designs with experimental designs also falls.

GL target	H overlap	L overlap	H edit	L edit
0.40	0.77	0.82	20.6	13.0
0.38	0.76	0.79	18.4	11.3

Table 7.0.5: Comparison of Humatch's designs when humanising the same 25 precursor therapeutics using different initial germline-likeness targets. 'H edit' and 'L edit' are the mean edit distances of the humanised therapeutics from their corresponding precursor sequences. 'H overlap' and 'L overlap' described the mean overlap in mutations made computationally compared to those made experimentally. If all suggested computational mutations were made experimentally, the overlap would be one; if none matched, the overlap would be zero.

Humatch humanisation of 25 precursor therapeutics allowing CDR mutations

CDR mutations can be allowed by the user in both stages of Humatch's humanisation pipeline: stage 1 - the initial germline-likeness matching, and stage 2 - the subsequent iterative design and selection of single-point mutants. By default, Humatch does not allow CDR mutations in either stage to increase its speed (fewer single-point variants require screening). Table 7.0.6 also shows that disallowing CDR mutations results in designs with greater experimental overlap by examining Humatch's performance when humanising 25 precursor therapeutics with known therapeutic endpoints. Nevertheless, if users do wish to allow mutations within the CDR regions, Figure 7.0.6 shows that these are likely to be small in number and located near the CDR anchors.

Stage 1	Stage 2	H overlap	L overlap	H edit	L edit
Disallow	Disallow	0.77	0.82	20.6	13.0
Allow	Disallow	0.73	0.80	21.2	13.0
Disallow	Allow	0.75	0.76	20.4	13.4
Allow	Allow	0.72	0.75	20.9	13.2

Table 7.0.6: Comparison of Humatch's designs when humanising the same 25 precursor therapeutics either allowing or disallowing CDR mutations in the two stages of Humatch's humanisation protocol. Stage 1 refers to the initial germline-likeness matching, and stage 2 refers to the iterative design and selection of single-point variants. 'H edit' and 'L edit' are the mean edit distances of the humanised therapeutics from their corresponding precursor sequences. 'H overlap' and 'L overlap' described the mean overlap in mutations made computationally compared to those made experimentally. If all suggested computational mutations were made experimentally, the overlap would be one; if none matched, the overlap would be zero.

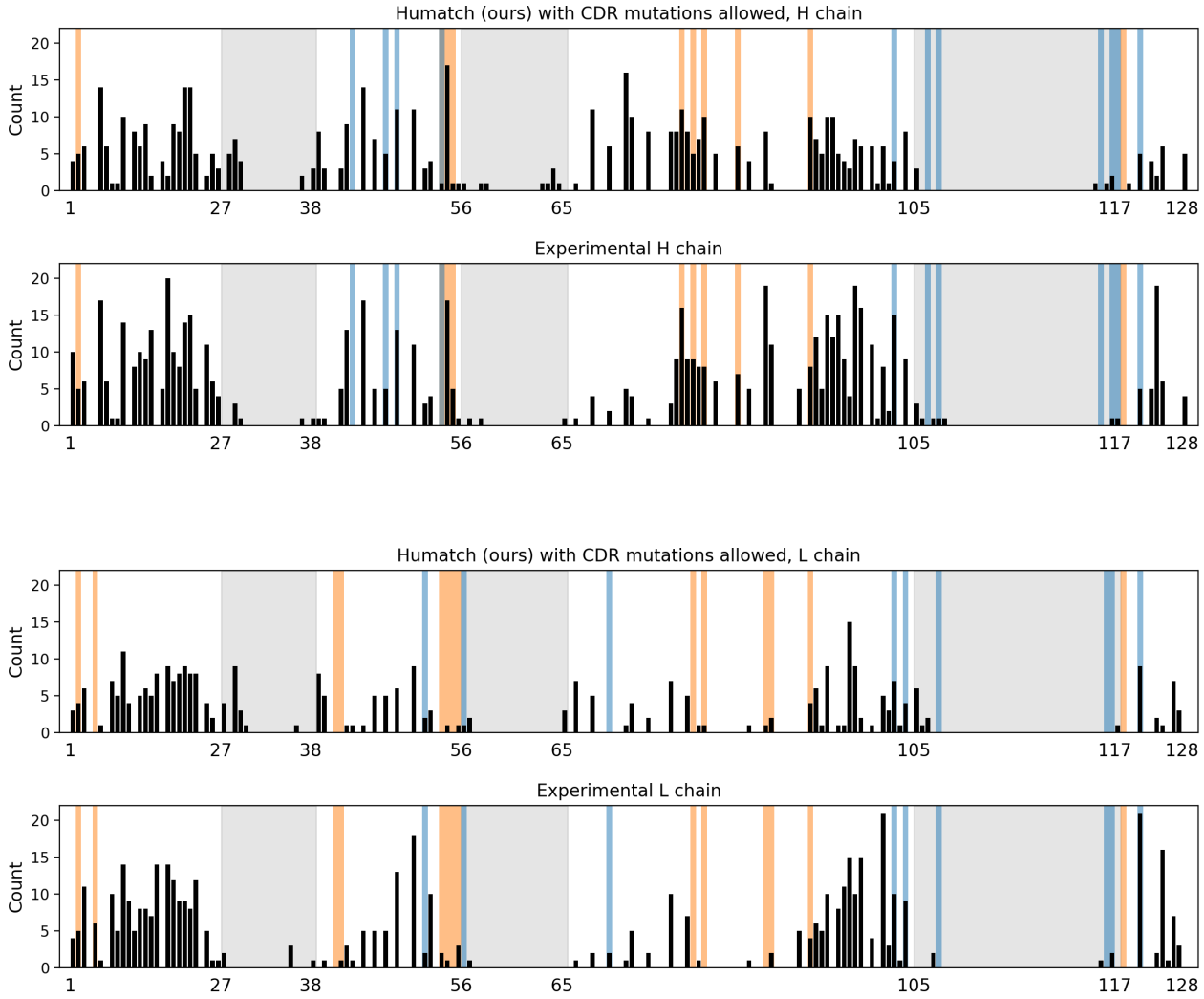

Figure 7.0.6: A comparison of Humatch and experimental mutation profiles for 25 precursor therapeutics, separated by heavy and light chains. Here, Humatch was run while allowing CDR mutations in both stages of its humanisation protocol. The black bars show the number of mutations made at each IMGT sequence position across the 25 therapeutics. CDR regions are shaded in grey, Vernier zone residues in orange, and interface residues in blue.

Nearest training data matches to 25 experimental endpoints

To check if Humatch simply memorised its training data and pushed designs towards these sequences during humanisation, we compared the 25 experimental endpoints to their closest training/validation matches (see Figure 7.0.7). The closest sequence matches were identified using KASearch with its pre-aligned version of OAS containing 2.4 billion sequences. We returned the closest 20 entries for every experimental heavy and light endpoint and then compared these against our training and validation data. As we did not include all OAS sequences in our training, some therapeutics returned no matches for all 20 hits (grey boxes in Figure 7.0.7). When hits were found, they were found to be distant, with mean heavy and light edit distances of 21 and 11 between the experimental endpoints and the closest training sequences. For comparison, we also include the edit distances between our designs and all experimental endpoints, and our designs and the closest experimental training hits in Figure 7.0.7. We see that our designs are closer to the experimental designs than their closest training dataset matches, indicating that Humatch does not simply drag designs towards sequences it has observed.

Therapeutic	H-edit (exp-ours)	L-edit (exp-ours)	H-edit (exp-closest)	L-edit (exp-closest)	H-edit (ours-closest)	L-edit (ours-closest)
AntiCD28	20	10	20	19	29	23
Bevacizumab	12	7	29	8	27	15
Campath	16	8		10		17
Certolizumab	13	11	25	13	28	17
Clazakizumab	11	6		14		19
Crizanlizumab	15	12	23	11	25	19
Eculizumab	16	4	22		28	
Etaracizumab	16	20	15	13	22	21
Herceptin	12	12		10		17
Idarucizumab	13	10	24		29	
Ixekizumab	21	10	19	11	26	15
Ligelizumab	16	14	24	11	30	16
Lorvotuzumab	6	8		8		16
Mogamulizumab	9	9	20	14	25	21
Omalizumab	15	11	25	12	29	20
Palivizumab	11	11				
Pembrolizumab	18	12				
Pertuzumab	13	10	23	12	24	16
Pinatuzumab	13	9	22		24	
Refanezumab	11	11	16	4	25	15
Reslizumab	16	11		11		16
Rovalpituzumab	14	14		10		18
Solanezumab	14	6	12	9	20	15
Talacotuzumab	23	10		8		16
Tocilizumab	15	7	20	9	21	16

Figure 7.0.7: Edit distances between our designs, experimental (exp) designs, and the training/validation sequences closest to the experimental designs for 25 humanised therapeutics. Results are split by heavy (H) and light (L) chains. Smaller edit distances are coloured in green and larger distances in red. The closest sequences were identified using KASearch and its pre-aligned version of OAS. As Humatch was not trained on all of OAS, some closest hits found using were not in the training data - these are shown as grey boxes.

Examining whether different tools humanise between genes

Figure 4.4.5 in Chapter 4 shows no evidence of any tool humanising 'between' genes according to Humatch's CNN-H and CNN-L classifiers. Figure 7.0.8 does not provide direct evidence for this either, but it does show that some tools' designs have germline content less clearly associated with only one V-gene. To avoid bias in this analysis, we did not use Humatch's classifications and instead only considered germline-likeness (GL) scores that measure how often amino acids are observed at different positions in sequences from OAS. First, we obtained the GL scores for every V-gene for all 25 designs from each tool. The first and second-highest GL scores for each sequence were then selected and their difference was calculated (ΔGL). Larger ΔGL values occur for sequences that share many commonalities with one V-gene as they are less able to match others. Smaller values indicate a sequence is similar to at least two V-genes. We observed that designs from Hu-mAb and AbNatiV generally had lower ΔGL scores. Designs from Sapiens and Humatch had higher ΔGL scores similar to those of experimentally humanised sequences. However, Humatch achieved these higher scores in fewer mutations than Sapiens and did so while targeting the same genes as experiments, unlike Sapiens.

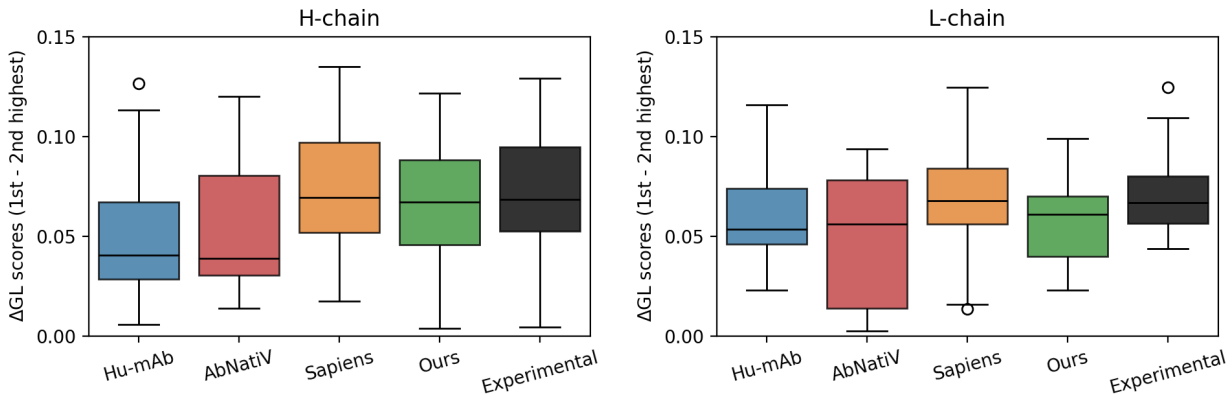

Figure 7.0.8: Boxplots showing the differences in the highest and second-highest germline-likeness (GL) scores for 25 humanised heavy and light designs from Hu-mAb, AbNatiV, Sapiens, Humatch, and experimental efforts.

Humatch classifies more 'mixed-gene' sequences as non-human than other tools

To test whether each humanisation tool can successfully identify 'mixed-gene' designs as non-human, we created a dataset of artificial mixed heavy V-gene sequences e.g. first half HV1, second half HV3 (see Figure 7.0.9). We created this dataset by sampling heavy V-gene sequences from Humatch's test set. We split sequences in the middle of the CDR2 (IMGT position 61E) as this is approximately the mid-point of the V-gene encoded region (IMGT positions 1 to 106).

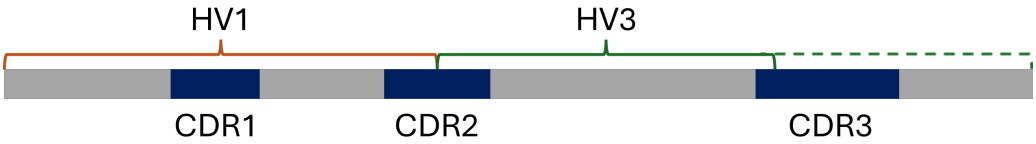

Figure 7.0.9: Overview of our mixed-gene designs - the start of the sequence, up to the middle of the CDR2 loop, is taken from one V-gene, while the rest of the sequence is from another.

As sequences were split in the middle of the CDR2, we only re-joined sequence halves that shared original CDR2 lengths. All heavy V-genes are dominated by a single-length CDR2 due to their germline origins: HV1&3&5&7 CDR2s are largely length-eight, HV2&4 are length-seven, and HV6 is length-nine. We ignored sequences with less common CDRH2 lengths resulting from insertions/deletions and all HV6 sequences, instead joining only those from HV1&3&5&7 together, and HV2&4 together. The resulting dataset - both the ∼57k single gene sequences sampled from Humatch's test set and the ∼122k mixed-gene creations - can be found at doi.org/10.5281/zenodo.13764770

Humatch, using a CNN-H cutoff of 0.95 (corresponding to near-perfect precision and 95.3% recall, see Table 7.0.2), successfully classified 51% of the 122k mixed-gene designs as non-human. OASis does not recommend a classifier cut-off in their paper so we instead calculated OASis Identity scores for our 57k single-gene sequences and determined the threshold (0.465) that also gave 95.3% recall. We found that only 1% of mixed-gene designs (fewer than true human sequences) fell below this threshold.

A similar process was followed for AbNatiV, giving a 95.3% recall threshold of 0.736. This cut-off removed 28% of mixed-gene designs, more than OASis but fewer than Humatch. AbNatiV

do however recommend a cut-off of 0.8 in their paper. This cut-off gives a lower recall of 83.3% on our 57k single-gene sequences but removes 67% of mixed-gene designs. At 83.3% recall (CNN-H cut-off of 0.99994), Humatch successfully classifies 94% of mixed-gene designs as non-human.

Finally, Hu-mAb was used to score the single and mixed gene sequences. Hu-mAb includes its own classification thresholds for each gene and, coincidentally, these provided an overall recall of 95.3% on our single-gene designs too. Using these default thresholds, Hu-mAb scored only 8% of mixed-gene designs as non-human. Furthermore, we found that 59% of mixed-gene designs were ranked as human according to multiple V-gene classifiers. This in itself is not enough to determine if a sequence is mixed-gene though as Hu-mAb also ranked 24% of the 57k single-gene designs as human according to multiple V-gene classifiers.

Note - OASis and Humatch scored all mixed-gene sequences as OASis does not number sequences and the sequences were pre-aligned for use by Humatch. However, Hu-mAb and AbNatiV both used ANARCI (Dunbar et al. 2016a) to number the sequences before scoring. ANARCI failed to number $\sim$2.2% of mixed-gene sequences so these were not scored by either tool.

Humanising 1,000 non-human sequences

In Chapter 4, we report the mean heavy and light edit distances and runtimes for the humanisation of 1,000 non-human sequences. Figure 7.0.10 shows the full distribution of values from this process. The majority of sequences are humanised in less than a minute and fewer than 40 edits.

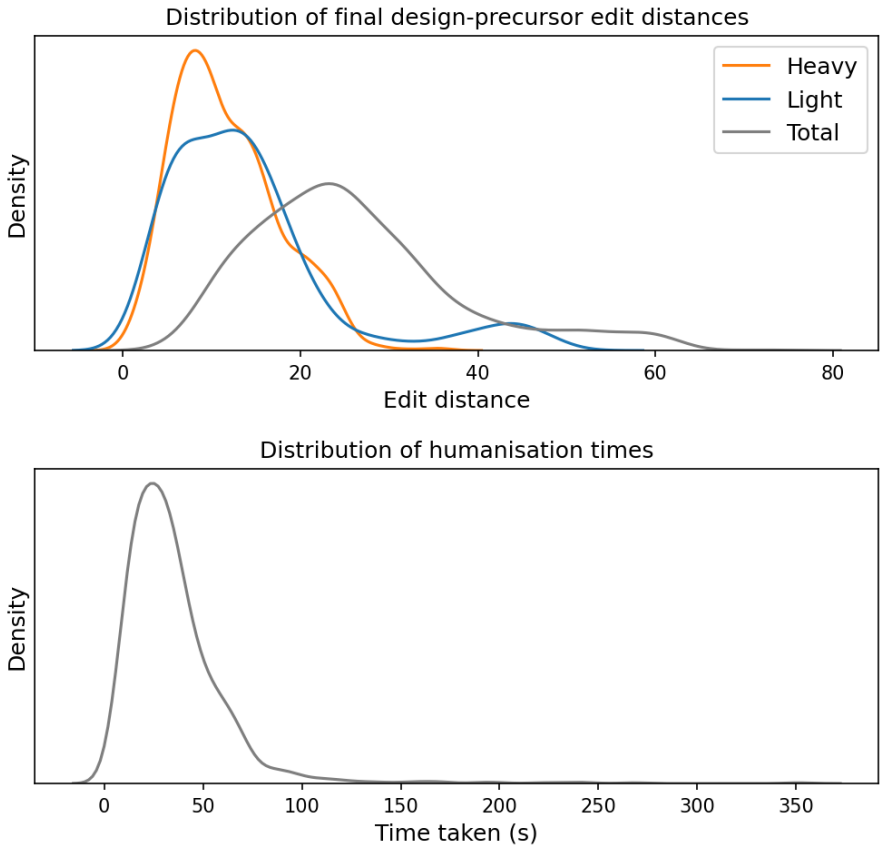

Figure 7.0.10: Distribution of heavy and light edit distances of 1,000 Humatch designs from their precursor non-human sequences, and the times taken to reach these designs.

Hu-mAb, AbNatiV, and Sapiens settings

We ran Hu-mab, AbNatiV, and Sapiens with default and/or paper-recommended parameters. Hu-mAb was run from SAbBox, a vagrant VirtualBox containing many SAbPred (Dunbar et al. 2016b) tools. Hu-mAb's 25 experimental heavy/light target scores and 211 ADA-therapeutic scores were obtained using e.g.

```
singularity exec /path/to/sabbox.sif Hu-mAb -h_seq <H_seq> -name
test -score_only
```

Sequences were then humanised using e.g.

```
singularity exec /path/to/sabbox.sif Hu-mAb -name <therapeutic>
-h_seq <H_seq> -l_seq <L_seq> -vgene_h <H_gene> -vgene_l <L_gene>
-threshold_h <H_target> -threshold_l <L_target> -v
```

AbNatiV's humanisation code was cloned from gitlab.developers.cam.ac.uk/ch/sormanni/abnativ. Sequences were scored using e.g.

```
abnativ score -nat VLambda -i <VL_fasta> -odir <VL_odir> -oid test
-align
```

Sequences were humanised using e.g.

```
abnativ hum_vh_vl -i_vh <H_seq> -i_vl <L_seq> -odir </out/dir>
-oid test_vh_vl
```

Sapiens and OASis were run from the web interface - biophi.dichlab.org. Sequences were input in fasta format, IMGT definitions were used for numbering and CDR definitions, and a relaxed OASis prevalence threshold of 10% was used for both scoring and humanisation. Sapiens was used for humanisation and three humanisation iterations were permitted. The CDRs were not allowed to be humanised.

References

Abanades, Brennan et al. (2022). "ABlooper: Fast accurate antibody CDR loop structure prediction with accuracy estimation". In: *Bioinformatics*. DOI: 10.1093/bioinformatics/btac016.

Abanades, Brennan et al. (May 2023). "ImmuneBuilder: Deep-Learning models for predicting the structures of immune proteins". In: *Communications Biology 2023 6:1* 6 (1), pp. 1–8. ISSN: 2399-3642. DOI: 10.1038/s42003-023-04927-7.

Abhinandan, K. R. et al. (Aug. 2008). "Analysis and improvements to Kabat and structurally correct numbering of antibody variable domains". In: *Molecular Immunology* 45 (14), pp. 3832–3839. ISSN: 0161-5890. DOI: 10.1016/J.MOLIMM.2008.05.022.

Abraham, Mark James et al. (Sept. 2015). "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers". In: *SoftwareX* 1-2, pp. 19–25. ISSN: 2352-7110. DOI: 10.1016/J.SOFTX.2015.06.001.

Ahmad, Zuhaida Asra et al. (2012). "scFv Antibody: Principles and Clinical Application". In: *Clinical and Developmental Immunology* 2012, p. 15. ISSN: 17402522. DOI: 10.1155/2012/980250.

Ahmed, Rafi et al. (1996). "Immunological Memory and Protective Immunity: Understanding Their Relation". In: *Science* 272.5258, pp. 54–60. DOI: 10.1126/science.272.5258.54.

Ain, Qurat ul et al. (Jan. 2020). "TLR4-Targeting Therapeutics: Structural Basis and Computer-Aided Drug Discovery Approaches". In: *Molecules 2020, Vol. 25, Page 627* 25 (3), p. 627. ISSN: 1420-3049. DOI: 10.3390/MOLECULES25030627.

Akbar, Rahmad et al. (2022). "Progress and challenges for the machine learning-based design of fit-for-purpose monoclonal antibodies". In: *mAbs* 14 (1). ISSN: 19420870. DOI: 10.1080/19420862.2021.2008790.

Alberts, B. et al. (2002). *Molecular Biology of the Cell*. 4th. Garland.

Almagro, Juan C. et al. (Sept. 2019). "Phage Display Libraries for Antibody Therapeutic Discovery and Development". In: *Antibodies* 8 (3). ISSN: 20734468. DOI: 10.3390/ANTIB8030044.

Ambrosetti, Francesco et al. (Jan. 2020a). "Modeling Antibody-Antigen Complexes by Information-Driven Docking". In: *Structure* 28 (1), 119–129.e2. ISSN: 0969-2126. DOI: 10.1016/J.STR.2019.10.011.

Ambrosetti, Francesco et al. (Dec. 2020b). "proABC-2: PRediction of AntiBody contacts v2 and its application to information-driven docking". In: *Bioinformatics* 36 (20), pp. 5107–5108. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/BTAA644.

Bachas, Sharrol et al. (Aug. 2022). "Antibody optimization enabled by artificial intelligence predictions of binding affinity and naturalness". In: *bioRxiv*, p. 2022.08.16.504181. DOI: 10.1101/2022.08.16.504181.

Bancroft, Tara et al. (Oct. 2019). "Detection and activation of HIV broadly neutralizing antibody precursor B cells using anti-idiotypes". In: *Journal of Experimental Medicine* 216 (10), pp. 2331–2347. ISSN: 0022-1007. DOI: 10.1084/JEM.20190164.

Bateman, Alex et al. (Jan. 2015). "UniProt: a hub for protein information". In: *Nucleic Acids Research* 43 (D1), pp. D204–D212. ISSN: 0305-1048. DOI: 10.1093/NAR/GKU989.

Bender, Brian J. et al. (Sept. 2021). "A practical guide to large-scale docking". In: *Nature Protocols 2021 16:10* 16 (10), pp. 4799–4832. ISSN: 1750-2799. DOI: 10.1038/s41596-021-00597-z.

Bernard, Ora et al. (Dec. 1978). "Sequences of mouse immunoglobulin light chain genes before and after somatic changes". In: *Cell* 15 (4), pp. 1133–1144. ISSN: 00928674. DOI: 10.1016/0092-8674(78)90041-7.

Bernett, Judith et al. (Aug. 2024). "Guiding questions to avoid data leakage in biological machine learning applications". In: *Nature Methods 2024 21:8* 21 (8), pp. 1444–1453. ISSN: 1548-7105. DOI: 10.1038/s41592-024-02362-y.

Boulot, G. et al. (July 1988). "Crystallization of antibody fragments and their complexes with antigen". In: *Journal of Crystal Growth* 90 (1-3), pp. 213–221. ISSN: 0022-0248. DOI: 10.1016/0022-0248(88)90318-1.

Brooks, B. R. et al. (July 2009). "CHARMM: The Biomolecular Simulation Program". In: *Journal of computational chemistry* 30 (10), p. 1545. ISSN: 1096987X. DOI: 10.1002/JCC.21287.

Brown, Tom B. et al. (May 2020). "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems* 2020-December. ISSN: 10495258. DOI: 10.48550/arXiv.2005.14165.

Buntz, Brian (2024). "Best-selling pharmaceuticals of 2023: Metabolic drugs shine". In: *Drug Discovery and Development*.

Burley, Stephen K. et al. (Jan. 2019). "RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy". In: *Nucleic Acids Research* 47 (D1), pp. D464–D474. ISSN: 0305-1048. DOI: 10.1093/NAR/GKY1004.

Burnet, F. M. (Mar. 1957). "A modification of jerne's theory of antibody production using the concept of clonal selection". In: *CA: A Cancer Journal for Clinicians* 26 (2), pp. 119–121. ISSN: 1542-4863. DOI: 10.3322/CANJCLIN.26.2.119.

Buttenschoen, Martin et al. (Feb. 2024). "PoseBusters: AI-based docking methods fail to generate physically valid poses or generalise to novel sequences". In: *Chemical Science* 15 (9), pp. 3130–3139. ISSN: 2041-6539. DOI: 10.1039/D3SC04185A.

Case, David A. et al. (Dec. 2005). "The Amber Biomolecular Simulation Programs". In: *Journal of computational chemistry* 26 (16), p. 1668. ISSN: 01928651. DOI: 10.1002/JCC.20290.

Chao, Ginger et al. (July 2006). "Isolating and engineering human antibodies using yeast surface display". In: *Nature Protocols 2006 1:2* 1 (2), pp. 755–768. ISSN: 1750-2799. DOI: 10.1038/nprot.2006.94.

Chinery, Lewis et al. (Jan. 2023). "Paragraph—antibody paratope prediction using graph neural networks with minimal feature vectors". In: *Bioinformatics* 39 (1). ISSN: 1367-4803. DOI: 10.1093/bioinformatics/BTAC732.

Chinery, Lewis et al. (Mar. 2024a). "Baselining the Buzz Trastuzumab-HER2 Affinity, and Beyond". In: *bioRxiv*, p. 2024.03.26.586756. DOI: 10.1101/2024.03.26.586756.

Chinery, Lewis et al. (Dec. 2024b). "Humatch - fast, gene-specific joint humanisation of antibody heavy and light chains". In: *mAbs* 16 (1). ISSN: 19420870. DOI: 10.1080/19420862.2024.2434121.

Cho, Hyun-Soo et al. (2003). "Structure of the extracellular region of HER2 alone and in complex with the Herceptin Fab". In: *Nature* 421.6924, pp. 756–760. DOI: 10.1038/nature01392.

Choi, Yoonjoo et al. (2010). "FREAD revisited: Accurate loop structure prediction using a database search algorithm". In: *Proteins* 78 (6), pp. 1431–1440. ISSN: 1097-0134. DOI: 10.1002/PROT.22658.

Chothia, Cyrus et al. (Aug. 1987). "Canonical structures for the hypervariable regions of immunoglobulins". In: *Journal of Molecular Biology* 196 (4), pp. 901–917. ISSN: 00222836. DOI: 10.1016/0022-2836(87)90412-8.

Crick, Francis et al. (1961). "General nature of the genetic code for proteins". In: *Nature 192, 1227–1232*. DOI: 10.1038/1921227a0.

Crotty, Shane et al. (Nov. 2003). "Cutting Edge: Long-Term B Cell Memory in Humans after Smallpox Vaccination". In: *The Journal of Immunology* 171 (10), pp. 4969–4973. ISSN: 0022-1767. DOI: 10.4049/JIMMUNOL.171.10.4969.

Daberdaku, Sebastian et al. (June 2019). "Antibody interface prediction with 3D Zernike descriptors and SVM". In: *Bioinformatics (Oxford, England)* 35 (11), pp. 1870–1876. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/BTY918.

Dauparas, Justas et al. (2022). "Robust deep learning–based protein sequence design using ProteinMPNN". In: *Science* 378.6615, pp. 49–56. DOI: 10.1126/science.add2187.

Devlin, Jacob et al. (Oct. 2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1, pp. 4171–4186.

Doerner, Achim et al. (Jan. 2014). "Therapeutic antibody engineering by high efficiency cell screening". In: *FEBS Letters* 588 (2), pp. 278–287. ISSN: 0014-5793. DOI: 10.1016/J.FEBSLET.2013.11.025.

Dondelinger, Mathieu et al. (Oct. 2018). "Understanding the significance and implications of antibody numbering and antigen-binding surface/residue definition". In: *Frontiers in Immunology* 9 (OCT), p. 412684. ISSN: 16643224. DOI: 10.3389/FIMMU.2018.02278.

Dunbar, James et al. (2014). "SAbDab: the structural antibody database". In: *Nucleic Acids Research*. DOI: 10.1093/nar/gkt1043.

Dunbar, James et al. (Jan. 2016a). "ANARCI: antigen receptor numbering and receptor classification". In: *Bioinformatics* 32 (2), pp. 298–300. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/BTV552.

Dunbar, James et al. (July 2016b). "SAbPred: a structure-based antibody prediction server". In: *Nucleic Acids Research* 44 (W1), W474–W478. ISSN: 0305-1048. DOI: 10.1093/NAR/GKW361.

Eberhardt, Jerome et al. (Aug. 2021). "AutoDock Vina 1.2.0: New Docking Methods, Expanded Force Field, and Python Bindings". In: *Journal of Chemical Information and Modeling* 61 (8), pp. 3891–3898. ISSN: 15205142. DOI: 10.1021/ACS.JCIM.1C00203.

Eddy, Sean R. (Aug. 2004). "Where did the BLOSUM62 alignment score matrix come from?" In: *Nature Biotechnology 2004 22:8* 22 (8), pp. 1035–1036. ISSN: 1546-1696. DOI: 10.1038/nbt0804-1035.

Evans, Richard et al. (Mar. 2022). "Protein complex prediction with AlphaFold-Multimer". In: *bioRxiv*, p. 2021.10.04.463034. DOI: 10.1101/2021.10.04.463034.

Ferrara, Fortunato et al. (Nov. 2012). "Using Phage and Yeast Display to Select Hundreds of Monoclonal Antibodies: Application to Antigen 85, a Tuberculosis Biomarker". In: *PLOS ONE* 7 (11), e49535. ISSN: 1932-6203. DOI: 10.1371/JOURNAL.PONE.0049535.

Foote, Jefferson et al. (Mar. 1992). "Antibody framework residues affecting the conformation of the hypervariable loops". In: *Journal of Molecular Biology* 224 (2), pp. 487–499. ISSN: 0022-2836. DOI: 10.1016/0022-2836(92)91010-M.

Frey, Nathan C et al. (June 2023). "Protein Discovery with Discrete Walk-Jump Sampling". In: *arXiv*. DOI: 10.48550/arXiv.2306.12360.

Fu, Limin et al. (Dec. 2012). "CD-HIT: accelerated for clustering the next-generation sequencing data". In: *Bioinformatics (Oxford, England)* 28 (23), pp. 3150–3152. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/BTS565.

Gartner, Leslie P. et al. (2011). "12 - Lymphoid (Immune) System". In: *Concise Histology*. Ed. by Leslie P. Gartner et al. Philadelphia: W.B. Saunders, pp. 168–187. ISBN: 978-0-7020-3114-4. DOI: 10.1016/B978-0-7020-3114-4.00012-9.

Gaston, Julie et al. (Dec. 2019). "Intracellular delivery of therapeutic antibodies into specific cells using antibody-peptide fusions". In: *Scientific Reports 2019 9:1* 9 (1), pp. 1–12. ISSN: 2045-2322. DOI: 10.1038/s41598-019-55091-0.

Gilmore, Eugene et al. (2021). "More Interpretable Decision Trees". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 12886 LNAI, pp. 280–292. ISSN: 16113349. DOI: 10.1007/978-3-030-86271-8_24.

Goldstein, Leonard D. et al. (Aug. 2019). "Massively parallel single-cell B-cell receptor sequencing enables rapid discovery of diverse antigen-reactive antibodies". In: *Communications Biology 2019 2:1* 2 (1), pp. 1–10. ISSN: 2399-3642. DOI: 10.1038/s42003-019-0551-y.

Gordon, Gemma L. et al. (May 2024). "Prospects for the computational humanization of antibodies and nanobodies". In: *Frontiers in Immunology* 15, p. 1399438. ISSN: 16643224. DOI: 10.3389/fimmu.2024.1399438.

Griffiths, Gillian M. et al. (1984). "Somatic mutation and the maturation of immune response to 2-phenyl oxazolone". In: *Nature* 312 (5991), pp. 271–275. ISSN: 0028-0836. DOI: 10.1038/312271A0.

Gupta, Sujata (Aug. 2017). "Trials and tribulations". In: *Nature 2017 548:7666* 548 (7666), S28–S31. ISSN: 1476-4687. DOI: 10.1038/548s28a.

Hegyi, Hedi et al. (Apr. 1999). "The relationship between protein structure and function: a comprehensive survey with application to the yeast genome". In: *Journal of Molecular Biology* 288 (1), pp. 147–164. ISSN: 0022-2836. DOI: 10.1006/JMBI.1999.2661.

Henikoff, S. et al. (Nov. 1992). "Amino acid substitution matrices from protein blocks." In: *Proceedings of the National Academy of Sciences* 89 (22), p. 10915. ISSN: 00278424. DOI: 10.1073/PNAS.89.22.10915.

Hie, Brian L. et al. (Apr. 2023). "Efficient evolution of human antibodies from general protein language models". In: *Nature Biotechnology 2023*, pp. 1–9. ISSN: 1546-1696. DOI: 10.1038/s41587-023-01763-2.

Hochreiter, Sepp et al. (Nov. 1997). "Long Short-Term Memory". In: *Neural Computation* 9 (8), pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/NECO.1997.9.8.1735.

Hoehn, Kenneth B. et al. (May 2016). "The Diversity and Molecular Evolution of B-Cell Receptors during Infection". In: *Molecular Biology and Evolution* 33 (5), p. 1147. ISSN: 15371719. DOI: 10.1093/MOLBEV/MSW015.

Honegger, Annemarie et al. (June 2001). "Yet Another Numbering Scheme for Immunoglobulin Variable Domains: An Automatic Modeling and Analysis Tool". In: *Journal of Molecular Biology* 309 (3), pp. 657–670. ISSN: 0022-2836. DOI: 10.1006/JMBI.2001.4662.

Honjo, T. (1983). "Immunoglobulin genes". In: *Annual review of immunology* 1, pp. 499–528. ISSN: 0732-0582. DOI: 10.1146/ANNUREV.IY.01.040183.002435.

Hoogenboom, Hennie R. (Sept. 2005). "Selecting and screening recombinant antibody libraries". In: *Nature Biotechnology 2005 23:9* 23 (9), pp. 1105–1116. ISSN: 1546-1696. DOI: 10.1038/nbt1126.

Hozumi, N. et al. (Oct. 1976). "Evidence for somatic rearrangement of immunoglobulin genes coding for variable and constant regions." In: *Proceedings of the National Academy of Sciences* 73 (10), pp. 3628–3632. ISSN: 00278424. DOI: 10.1073/PNAS.73.10.3628.

Hu, Yunfei et al. (Feb. 2021). "NMR-Based Methods for Protein Analysis". In: *Analytical Chemistry* 93 (4), pp. 1866–1879. ISSN: 15206882. DOI: 10.1021/acs.analchem.0c03830.

Hughes, J. P. et al. (Mar. 2011). "Principles of early drug discovery". In: *British Journal of Pharmacology* 162 (6), p. 1239. ISSN: 00071188. DOI: 10.1111/J.1476-5381.2010.01127.X.

Hummer, Alissa M. et al. (June 2022). "Advances in computational structure-based antibody design". In: *Current Opinion in Structural Biology* 74, p. 102379. ISSN: 0959-440X. DOI: 10.1016/J.SBI.2022.102379.

Hummer, Alissa M. et al. (2023). "Investigating the Volume and Diversity of Data Needed for Generalizable Antibody-Antigen G Prediction". In: *bioRxiv*. DOI: 10.1101/2023.05.17.541222.

Hwang, William Ying Khee et al. (May 2005). "Immunogenicity of engineered antibodies". In: *Methods* 36 (1), pp. 3–10. ISSN: 1046-2023. DOI: 10.1016/J.YMETH.2005.01.001.

Høie, Magnus Haraldson et al. (May 2024). "AntiFold: Improved antibody structure-based design using inverse folding". In: *NeurIPS 2023 Generative AI and Biology (GenBio) Workshop*, pp. 1–17.

Jain, Tushar et al. (Jan. 2017). "Biophysical properties of the clinical-stage antibody landscape". In: *Proceedings of the National Academy of Sciences* 114 (5), pp. 944–949. ISSN: 10916490. DOI: 10.1073/PNAS.1616408114.

Janeway, C. A. (Jan. 1989). "Approaching the Asymptote? Evolution and Revolution in Immunology". In: *Cold Spring Harbor Symposia on Quantitative Biology* 54 (1), pp. 1–13. ISSN: 0091-7451. DOI: 10.1101/SQB.1989.054.01.003.

Janeway, Jr Charles A et al. (2001). "Immunobiology". In: *Immunobiology* (14102), pp. 1–10.

Jenner, E. (1798). *An inquiry into the causes and effects of the variolae vaccinae.*

Jin, Shijie et al. (Feb. 2022). "Emerging new therapeutic antibody derivatives for cancer treatment". In: *Signal Transduction and Targeted Therapy 2022 7:1* 7 (1), pp. 1–28. ISSN: 2059-3635. DOI: 10.1038/s41392-021-00868-x.

Jones, Peter T. et al. (1986). "Replacing the complementarity-determining regions in a human antibody with those from a mouse". In: *Nature 1986 321:6069* 321 (6069), pp. 522–525. ISSN: 1476-4687. DOI: 10.1038/321522a0.

Jumper, John et al. (July 2021). "Highly accurate protein structure prediction with AlphaFold". In: *Nature 2021 596:7873* 596 (7873), pp. 583–589. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03819-2.

Kabat, EA (1983). *Sequences of proteins of immunological interest.* U.S. Dept. of Health and Human Services, Public Health Service, National Institutes of Health, Bethesda, MD.

Kamat, Vishal et al. (Nov. 2017). "Designing binding kinetic assay on the bio-layer interferometry (BLI) biosensor to characterize antibody-antigen interactions". In: *Analytical Biochemistry* 536, pp. 16–31. ISSN: 0003-2697. DOI: 10.1016/J.AB.2017.08.002.

Kantarjian, Hagop et al. (Mar. 2017). "Blinatumomab versus Chemotherapy for Advanced Acute Lymphoblastic Leukemia". In: *New England Journal of Medicine* 376 (9), pp. 836–847. ISSN: 0028-4793. DOI: 10.1056/NEJMoa1609783.

Kapoor, Sayash et al. (Sept. 2023). "Leakage and the reproducibility crisis in machine-learning-based science". In: *Patterns* 4 (9). ISSN: 26663899. DOI: 10.1016/J.PATTER.2023.100804.

Keck, Mei Le et al. (May 2018). "Mapping Determinants of Virus Neutralization and Viral Escape for Rational Design of a Hepatitis C Virus Vaccine". In: *Frontiers in Immunology* 9, p. 370510. ISSN: 16643224. DOI: 10.3389/FIMMU.2018.01194/BIBTEX.

Ketata, Mohamed Amine et al. (Apr. 2023). "DiffDock-PP: Rigid Protein-Protein Docking with Diffusion Models". In: *arXiv*. DOI: 10.48550/arXiv.2304.03889.

Khani-Habibabadi, Fatemeh et al. (2024). "Immunoglobulins; Fundamentals and Their Role in Neurological Disease". In: *Reference Module in Neuroscience and Biobehavioral Psychology*. DOI: 10.1016/B978-0-323-95702-1.00066-X.

Kidera, Akinori et al. (Feb. 1985). "Statistical analysis of the physical properties of the 20 naturally occurring amino acids". In: *Journal of Protein Chemistry* 4 (1), pp. 23–55. ISSN: 02778033. DOI: 10.1007/BF01025492/METRICS.

Kirschning, Carsten J. et al. (Dec. 1998). "Human Toll-like Receptor 2 Confers Responsiveness to Bacterial Lipopolysaccharide". In: *Journal of Experimental Medicine* 188 (11), pp. 2091–2097. ISSN: 0022-1007. DOI: 10.1084/JEM.188.11.2091.

Ko, Sanghwan et al. (Mar. 2021). "Recent Achievements and Challenges in Prolonging the Serum Half-Lives of Therapeutic IgG Antibodies Through Fc Engineering". In: *BioDrugs* 35 (2), pp. 147–157. ISSN: 1179190X. DOI: 10.1007/S40259-021-00471-0.

Kovaltsuk, Aleksandr et al. (Oct. 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. ISSN: 0022-1767. DOI: 10.4049/JIMMUNOL.1800708.

Lee, Jae Hyeon et al. (2023). "EquiFold: Protein Structure Prediction with a Novel Coarse-Grained Structure Representation". In: *bioRxiv*. DOI: 10.1101/2022.10.07.511322.

Leem, Jinwoo et al. (2016). "ABodyBuilder: Automated antibody structure prediction with data-driven accuracy estimation". In: *mAbs*. DOI: 10.1080/19420862.2016.1205773.

Lefranc, Marie Paule et al. (Jan. 2003). "IMGT unique numbering for immunoglobulin and T cell receptor variable domains and Ig superfamily V-like domains". In: *Developmental and Comparative Immunology* 27 (1), pp. 55–77. ISSN: 0145305X. DOI: 10.1016/S0145-305X(02)00039-3.

Levy, N. S. et al. (June 1989). "Early onset of somatic mutation in immunoglobulin VH genes during the primary immune response". In: *The Journal of Experimental Medicine* 169 (6), p. 2007. ISSN: 00221007. DOI: 10.1084/JEM.169.6.2007.

Li, Lin et al. (June 2023). "Machine learning optimization of candidate antibody yields highly diverse sub-nanomolar affinity antibody libraries". In: *Nature Communications 2023 14:1* 14 (1), pp. 1–12. ISSN: 2041-1723. DOI: 10.1038/s41467-023-39022-2.

Li, Weizhong et al. (Mar. 2001). "Clustering of highly homologous sequences to reduce the size of large protein databases". In: *Bioinformatics* 17 (3), pp. 282–283. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/17.3.282.

Liberis, Edgar et al. (2018). "Parapred: antibody paratope prediction using convolutional and recurrent neural networks". In: *Bioinformatics*. DOI: 10.1093/bioinformatics/bty305.

Linderstrom-Lang, K. U. (1952). *Lane Medical Lectures*. Standford University Press.

Lu, Ruei Min et al. (Jan. 2020). "Development of therapeutic antibodies for the treatment of diseases". In: *Journal of Biomedical Science 2020 27:1* 27 (1), pp. 1–30. ISSN: 1423-0127. DOI: 10.1186/S12929-019-0592-Z.

LucidRains, Phil Wang (2021). "EGNN - Pytorch". In: *GitHub*. DOI: github.com/lucidrains/egnn-pytorch.

López-Santibáñez-Jácome, Laura et al. (Apr. 2019). "The pipeline repertoire for Ig-Seq analysis". In: *Frontiers in Immunology* 10 (APR), p. 445111. ISSN: 16643224. DOI: 10.3389/FIMMU.2019.00899.

MacCallum, Robert M. et al. (Oct. 1996). "Antibody-antigen Interactions: Contact Analysis and Binding Site Topography". In: *Journal of Molecular Biology* 262 (5), pp. 732–745. ISSN: 0022-2836. DOI: 10.1006/JMBI.1996.0548.

MacLennan, Ian C.M. (Apr. 1994). "Germinal centers". In: *Annual Review of Immunology* 12 (Volume 12, 1994), pp. 117–139. ISSN: 07320582. DOI: 10.1146/ANNUREV.IY.12.040194.001001.

Malia, Thomas J. et al. (Oct. 2011). "Crystallization of a challenging antigen–antibody complex: TLR3 ECD with three noncompeting Fabs". In: *Acta Crystallographica Section F: Structural Biology and Crystallization Communications* 67 (Pt 10), p. 1290. ISSN: 17443091. DOI: 10.1107/S1744309111030983.

Marks, Claire et al. (Nov. 2021). "Humanization of antibodies using a machine learning approach on large-scale repertoire data". In: *Bioinformatics* 37 (22), pp. 4041–4047. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/BTAB434.

Mason, Derek M. et al. (Apr. 2021). "Optimization of therapeutic antibodies by predicting antigen specificity from antibody sequence via deep learning". In: *Nature Biomedical Engineering 2021 5:6* 5 (6), pp. 600–612. ISSN: 2157-846X. DOI: 10.1038/s41551-021-00699-9.

McDermott, Matthew B. A. et al. (Jan. 2024). "A Closer Look at AUROC and AUPRC under Class Imbalance". In: *arXiv*. DOI: 10.48550/arXiv.2401.06091.

Meyer-Baese, Anke et al. (2014). "Pattern Recognition and Signal Analysis in Medical Imaging: Second Edition". In: *Pattern Recognition and Signal Analysis in Medical Imaging: Second Edition*, pp. 1–444. DOI: 10.1016/C2012-0-00347-X.

Milholland, Brandon et al. (May 2017). "Differences between germline and somatic mutation rates in humans and mice". In: *Nature Communications 2017 8:1* 8 (1), pp. 1–8. ISSN: 2041-1723. DOI: 10.1038/ncomms15183.

Mullard, Asher (July 2021). "FDA approves 100th monoclonal antibody product". In: *Nature reviews. Drug discovery* 20 (7), pp. 491–495. ISSN: 14741784. DOI: 10.1038/D41573-021-00079-7.

Olsen, Tobias H et al. (Jan. 2022a). "AbLang: an antibody language model for completing antibody sequences". In: *Bioinformatics Advances* 2 (1). DOI: 10.1093/BIOADV/VBAC046.

Olsen, Tobias H. et al. (Jan. 2022b). "Observed Antibody Space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences". In: *Protein Science* 31 (1), pp. 141–146. ISSN: 1469896X. DOI: 10.1002/PRO.4205.

Olsen, Tobias H. et al. (July 2023). "KA-Search, a method for rapid and exhaustive sequence identity search of known antibodies". In: *Scientific Reports 2023 13:1* 13 (1), pp. 1–11. ISSN: 2045-2322. DOI: 10.1038/s41598-023-38108-7.

Olsen, Tobias H. et al. (Feb. 2024). "Addressing the antibody germline bias and its effect on language models for improved antibody design". In: *bioRxiv*, p. 2024.02.02.578678. DOI: 10.1101/2024.02.02.578678.

O'Shea, John J. et al. (Apr. 2008). "Cytokine Signaling Modules in Inflammatory Responses". In: *Immunity* 28 (4), pp. 477–487. ISSN: 1074-7613. DOI: 10.1016/J.IMMUNI.2008.03.002.

Paul, Steven M. et al. (Feb. 2010). "How to improve RD productivity: the pharmaceutical industry's grand challenge". In: *Nature Reviews Drug Discovery 2010 9:3* 9 (3), pp. 203–214. ISSN: 1474-1784. DOI: 10.1038/nrd3078.

Pedersen, Jan T. et al. (Jan. 1994). "Comparison of surface accessible residues in human and murine immunoglobulin Fv domains. Implication for humanization of murine antibodies". In: *Journal of molecular biology* 235 (3), pp. 959–973. ISSN: 0022-2836. DOI: 10.1006/JMBI.1994.1050.

Phillips, Angela M. et al. (Sept. 2021). "Binding affinity landscapes constrain the evolution of broadly neutralizing anti-influenza antibodies". In: *eLife* 10. ISSN: 2050084X. DOI: 10.7554/ELIFE.71393.

Phillips, James C. et al. (Dec. 2005). "Scalable molecular dynamics with NAMD". In: *Journal of Computational Chemistry* 26 (16), pp. 1781–1802. ISSN: 1096-987X. DOI: 10.1002/JCC.20289.

Pierce, Brian G. et al. (June 2014). "ZDOCK server: interactive docking prediction of protein–protein complexes and symmetric multimers". In: *Bioinformatics* 30 (12), p. 1771. DOI: 10.1093/bioinformatics/BTU097.

Pittala, Srivamshi et al. (July 2020). "Learning context-aware structural representations to predict antigen and antibody binding interfaces". In: *Bioinformatics* 36 (13), pp. 3996–4003. ISSN: 14602059. DOI: 10.1093/bioinformatics/btaa263.

Porebski, Benjamin T. et al. (Oct. 2023). "Rapid discovery of high-affinity antibodies via massively parallel sequencing, ribosome display and affinity screening". In: *Nature Biomedical Engineering 2023*, pp. 1–19. ISSN: 2157-846X. DOI: 10.1038/s41551-023-01093-3.

Prihoda, David et al. (Dec. 2022). "BioPhi: A platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning". In: *mAbs* 14 (1). ISSN: 19420870. DOI: 10.1080/19420862.2021.2020203.

Rabia, Lilia A. et al. (Sept. 2018). "Understanding and overcoming trade-offs between antibody affinity, specificity, stability and solubility". In: *Biochemical Engineering Journal* 137, p. 365. ISSN: 1873295X. DOI: 10.1016/J.BEJ.2018.06.003.

Ramon, Aubin et al. (Jan. 2024). "Assessing antibody and nanobody nativeness for hit selection and humanization with AbNatiV". In: *Nature Machine Intelligence 2024 6:1* 6 (1), pp. 74–91. ISSN: 2522-5839. DOI: 10.1038/s42256-023-00778-3.

Raybould, Matthew I.J. et al. (Mar. 2019). "Five computational developability guidelines for therapeutic antibody profiling". In: *Proceedings of the National Academy of Sciences* 116 (10), pp. 4025–4030. ISSN: 10916490. DOI: 10.1073/PNAS.1810576116.

Raybould, Matthew I.J. et al. (Jan. 2020). "Thera-SAbDab: the Therapeutic Structural Antibody Database". In: *Nucleic Acids Research* 48 (D1), pp. D383–D388. ISSN: 0305-1048. DOI: 10.1093/NAR/GKZ827.

Richardson, Eve et al. (June 2024). "The receiver operating characteristic curve accurately assesses imbalanced datasets". In: *Patterns* 5 (6). ISSN: 26663899. DOI: 10.1016/j.patter.2024.100994.

Riechmann, Lutz et al. (1988). "Reshaping human antibodies for therapy". In: *Nature 1988 332:6162* 332 (6162), pp. 323–327. ISSN: 1476-4687. DOI: 10.1038/332323a0.

Rives, Alexander et al. (Apr. 2021). "Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences". In: *Proceedings of the National Academy of Sciences* 118 (15). ISSN: 0027-8424. DOI: 10.1073/PNAS.2016239118.

Roguska, Michael A et al. (1994). "Humanization of Murine Monoclonal Antibodies Through Variable Domain Resurfacing". In: *Proceedings of the National Academy of Sciences* 91 (3), pp. 969–973. DOI: 10.1073/pnas.91.3.969.

Roskos, Lorin K. et al. (Mar. 2004). "The clinical pharmacology of therapeutic monoclonal antibodies". In: *Drug Development Research* 61 (3), pp. 108–120. ISSN: 1098-2299. DOI: 10.1002/DDR.10346.

Roth, David B. (Nov. 2014). "V(D)J Recombination: Mechanism, Errors, and Fidelity". In: *Microbiology spectrum* 2 (6). ISSN: 2165-0497. DOI: 10.1128/MICROBIOLSPEC.MDNA3-0041-2014.

Ruffolo, Jeffrey A. et al. (Apr. 2023). "Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies". In: *Nature Communications 2023 14:1* 14 (1), pp. 1–13. ISSN: 2041-1723. DOI: 10.1038/s41467-023-38063-x.

Satorras, Victor Garcia et al. (2021). "E(n) Equivariant Graph Neural Networks". In: *arXiv*. DOI: 10.48550/arXiv.2102.09844.

Schardt, John S. et al. (July 2024). "Monoclonal Antibody Generation Using Single B Cell Screening for Treating Infectious Diseases". In: *BioDrugs* 38 (4), pp. 477–486. ISSN: 1179190X. DOI: 10.1007/S40259-024-00667-0.

Schneider, Constantin et al. (Jan. 2022). "SAbDab in the age of biotherapeutics: updates including SAbDab-nano, the nanobody structure tracker". In: *Nucleic Acids Research* 50 (D1), pp. D1368–D1372. ISSN: 0305-1048. DOI: 10.1093/NAR/GKAB1050.

Schymkowitz, Joost et al. (2005). "The FoldX web server: An online force field". In: *Nucleic Acids Research* 33.SUPPL. 2, pp. 382–388. ISSN: 03051048. DOI: 10.1093/nar/gki387.

Seydoux, Emilie et al. (May 2021). "Development of a VRC01-class germline targeting immunogen derived from anti-idiotypic antibodies". In: *Cell Reports* 35 (5). ISSN: 22111247. DOI: 10.1016/j.celrep.2021.109084.

Shanehsazzadeh, Amir et al. (Mar. 2023). "Unlocking de novo antibody design with generative artificial intelligence". In: *bioRxiv*, p. 2023.01.08.523187. DOI: 10.1101/2023.01.08.523187.

Shannon, Michele et al. (Apr. 1999). "Reconciling Repertoire Shift with Affinity Maturation: The Role of Deleterious Mutations". In: *The Journal of Immunology* 162 (7), pp. 3950–3956. ISSN: 0022-1767. DOI: 10.4049/JIMMUNOL.162.7.3950.

Shuai, Richard W et al. (2021). "Generative language modeling for antibody design". In: *bioRxiv*, pp. 2021–12. DOI: 10.1101/2021.12.13.472419.

Sievers, Eric L. et al. (Jan. 2013). "Antibody-drug conjugates in cancer therapy". In: *Annual Review of Medicine* 64 (Volume 64, 2013), pp. 15–29. ISSN: 00664219. DOI: 10.1146/ANNUREV-MED-050311-201823.

Solaiman, Irene et al. (2019). "OpenAI Report Release Strategies and the Social Impacts of Language Models". In: *arXiv*. DOI: 10.48550/arXiv.1908.09203.

Starr, Tyler N. et al. (Feb. 2021). "Prospective mapping of viral mutations that escape antibodies used to treat COVID-19". In: *Science* 371 (6531), pp. 850–854. ISSN: 10959203. DOI: 10.1126/SCIENCE.ABF9302.

Stoeckius, Marlon et al. (July 2017). "Simultaneous epitope and transcriptome measurement in single cells". In: *Nature Methods 2017 14:9* 14 (9), pp. 865–868. ISSN: 1548-7105. DOI: 10.1038/nmeth.4380.

Tan, Yann Chong et al. (Oct. 2014). "Barcode-enabled sequencing of plasmablast antibody repertoires in rheumatoid arthritis". In: *Arthritis & Rheumatology (Hoboken, N.J.)* 66 (10), pp. 2706–2715. ISSN: 2326-5205. DOI: 10.1002/ART.38754.

Tauber, Alfred I. (Nov. 2003). "Metchnikoff and the phagocytosis theory". In: *Nature Reviews Molecular Cell Biology 2003 4:11* 4 (11), pp. 897–901. ISSN: 1471-0080. DOI: 10.1038/nrm1244.

Tiller, Kathryn E. et al. (Dec. 2015). "Advances in Antibody Design". In: *Annual Review of Biomedical Engineering* 17 (Volume 17, 2015), pp. 191–216. ISSN: 15454274. DOI: 10.1146/ANNUREV-BIOENG-071114-040733.

Tonegawa, Susumu (1983). "Somatic generation of antibody diversity". In: *Nature 1983 302:5909* 302 (5909), pp. 575–581. ISSN: 1476-4687. DOI: 10.1038/302575a0.

Torre, Beatriz G. de la et al. (Feb. 2024). "The Pharmaceutical Industry in 2023: An Analysis of FDA Drug Approvals from the Perspective of Molecules". In: *Molecules* 29 (3). ISSN: 14203049. DOI: 10.3390/MOLECULES29030585.

Touvron, Hugo et al. (Feb. 2023). "LLaMA: Open and Efficient Foundation Language Models". In: *arXiv*. DOI: 10.48550/arXiv.2302.13971.

Trabolsi, Asaad et al. (Aug. 2019). "T Cell–Activating Bispecific Antibodies in Cancer Therapy". In: *The Journal of Immunology* 203 (3), pp. 585–592. ISSN: 0022-1767. DOI: 10.4049/JIMMUNOL.1900496.

Urquhart, Lisa (Mar. 2019). "Top drugs and companies by sales in 2018". In: *Nature Reviews Drug Discovery*. ISSN: 1474-1776. DOI: 10.1038/D41573-019-00049-0.

Van Durme, Joost et al. (Apr. 2011). "A graphical interface for the FoldX forcefield". In: *Bioinformatics* 27.12, pp. 1711–1712. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btr254.

Vaswani, Ashish et al. (June 2017). "Attention Is All You Need". In: *Advances in Neural Information Processing Systems* 2017-December, pp. 5999–6009. ISSN: 10495258. DOI: 10.48550/arxiv.1706.03762.

Vecchio, Alice Del et al. (2021). "Neural message passing for joint paratope-epitope prediction". In: *arXiv*. DOI: 10.48550/arXiv.2106.00757.

Wang, Bo et al. (Jan. 2021). "Optimization of therapeutic antibodies". In: *Antibody Therapeutics* 4 (1), p. 45. ISSN: 25164236. DOI: 10.1093/ABT/TBAB003.

Wang, Chi et al. (2020). "FLAML: A Fast and Lightweight AutoML Library". In: *arXiv*. DOI: 10.48550/arXiv.1911.04706.

Wang, Hong Wei et al. (Jan. 2017). "How cryo-electron microscopy and X-ray crystallography complement each other". In: *Protein Science : A Publication of the Protein Society* 26 (1), p. 32. ISSN: 1469896X. DOI: 10.1002/PRO.3022.

Wen, Dingyi et al. (May 2013). "Discovery and investigation of O-xylosylation in engineered proteins containing a (GGGGS)n linker". In: *Analytical Chemistry* 85 (9), pp. 4805–4812. ISSN: 00032700. DOI: 10.1021/AC400596G.

Whitehead, Timothy A. et al. (May 2012). "Optimization of affinity, specificity and function of designed influenza inhibitors using deep sequencing". In: *Nature Biotechnology 2012 30:6* 30 (6), pp. 543–548. ISSN: 1546-1696. DOI: 10.1038/nbt.2214.

Williams, David Gareth et al. (2010). "Humanising Antibodies by CDR Grafting". In: *Antibody Engineering*, pp. 319–339. ISSN: 1949-2456. DOI: 10.1007/978-3-642-01144-3_21.

Williams, La Tonya D. et al. (Jan. 2017). "Potent and broad HIV-neutralizing antibodies in memory B cells and plasma". In: *Science Immunology* 2 (7). ISSN: 24709468. DOI: 10.1126/SCIIMMUNOL.AAL2200.

Williams, Nathan P. et al. (Jan. 2023). "DockNet: high-throughput protein–protein interface contact prediction". In: *Bioinformatics* 39 (1). ISSN: 1367-4803. DOI: 10.1093/bioinformatics/BTAC797.

Winter, Greg et al. (1991). "Man-made antibodies". In: *Nature 1991 349:6307* 349 (6307), pp. 293–299. ISSN: 1476-4687. DOI: 10.1038/349293a0.

Wouters, Olivier J. et al. (Mar. 2020). "Estimated Research and Development Investment Needed to Bring a New Medicine to Market, 2009-2018". In: *JAMA* 323 (9), pp. 844–853. ISSN: 1538-3598. DOI: 10.1001/JAMA.2020.1166.

Wright, John W. et al. (Jan. 1995). "The Angiotensin IV System: Functional Implications". In: *Frontiers in Neuroendocrinology* 16 (1), pp. 23–52. ISSN: 0091-3022. DOI: 10.1006/FRNE.1995.1002.

Xiao, Xiao et al. (Nov. 2019). "Characterization of potent RSV neutralizing antibodies isolated from human memory B cells and identification of diverse RSV/hMPV cross-neutralizing epitopes". In: *mAbs* 11 (8), pp. 1415–1427. ISSN: 19420870. DOI: 10.1080/19420862.2019.1654304.

Yaari, Gur et al. (Nov. 2015). "Practical guidelines for B-cell receptor repertoire sequencing analysis". In: *Genome Medicine 2015 7:1* 7 (1), pp. 1–14. ISSN: 1756-994X. DOI: 10.1186/S13073-015-0243-2.

Yang, Jianying et al. (2016). "Receptor Dissociation and B-Cell Activation". In: *B Cell Receptor Signaling*. Ed. by Tomohiro Kurosaki et al. Cham: Springer International Publishing, pp. 27–43. ISBN: 978-3-319-26133-1. DOI: 10.1007/82_2015_482.

Yang, Ruey Bing et al. (Sept. 1998). "Toll-like receptor-2 mediates lipopolysaccharide-induced cellular signalling". In: *Nature 1998 395:6699* 395 (6699), pp. 284–288. ISSN: 1476-4687. DOI: 10.1038/26239.

Youden, W. J. (1950). "Index for rating diagnostic tests". In: *Cancer* 3.1, pp. 32–35. DOI: 10.1002/1097-0142.

Yu, Xiaocong et al. (Aug. 2008). "Neutralizing antibodies derived from the B cells of 1918 influenza pandemic survivors". In: *Nature 2008 455:7212* 455 (7212), pp. 532–536. ISSN: 1476-4687. DOI: 10.1038/nature07231.

Zundert, G. C.P. Van et al. (Feb. 2016). "The HADDOCK2.2 Web Server: User-Friendly Integrative Modeling of Biomolecular Complexes". In: *Journal of Molecular Biology* 428 (4), pp. 720–725. ISSN: 0022-2836. DOI: 10.1016/J.JMB.2015.09.014.