FAST AND ACCURATE ANTIBODY SEQUENCE DESIGN VIA STRUCTURE RETRIEVAL

Xingyi Zhang[1*] **Kun Xie**[2,3*†] **Ningqiao Huang**[2] **Wei Liu**[2] **Peilin Zhao**[2]
Sibo Wang[3] **Kangfei Zhao**[4‡] **Biaobin Jiang**[2‡]
[1]MBZUAI [2]Tencent AI for Life Sciences Lab [3]The Chinese University of Hong Kong
[4]Beijing Institute of Technology

ABSTRACT

Recent advancements in protein design have leveraged diffusion models to generate structural scaffolds, followed by a process known as protein inverse folding, which involves sequence inference on these scaffolds. However, these methodologies face significant challenges when applied to hyper-variable structures such as antibody Complementarity-Determining Regions (CDRs), where sequence inference frequently results in non-functional sequences due to hallucinations. Distinguished from prevailing protein inverse folding approaches, this paper introduces IgSeek, a novel structure-retrieval framework that infers CDR sequences by retrieving similar structures from a natural antibody database. Specifically, IgSeek employs a simple yet effective multi-channel equivariant graph neural network to generate high-quality geometric representations of CDR backbone structures. Subsequently, it aligns sequences of structurally similar CDRs and utilizes structurally conserved sequence motifs to enhance inference accuracy. Our experiments demonstrate that IgSeek not only proves to be highly efficient in structural retrieval but also outperforms state-of-the-art approaches in sequence recovery for both antibodies and T-Cell Receptors, offering a new retrieval-based perspective for therapeutic protein design.

1 MAIN

Antibodies, known for their high specificity and affinity, have emerged as pivotal therapeutic agents in the treatment of complex diseases, including cancer Adams & Weiner (2005), autoimmune disorders Feldmann & Maini (2003), and infectious diseases Abraham (2020). In 2023, the global best-selling drug was Keytruda, a cancer treatment antibody, with sales reaching $25 billion, surpassing Humira, another antibody used for treating rheumatoid arthritis, which had dominated the market for the past decade (Dunleavy, 2024). Traditionally, the discovery of antibodies has predominantly relied on immunizing animals with antigens Van Wauwe et al. (1980) or employing various display techniques such as phage MacCallum et al. (1996) and yeast displays Chao et al. (2006). However, these approaches face significant challenges when dealing with structurally intricate proteins, which are difficult to express in a soluble and functional form. Additionally, even when numerous candidate antibodies are generated through these techniques, they may not necessarily bind to the desired domain or exhibit therapeutic efficacy.

To overcome these limitations, deep learning models have been introduced to design synthetic antibodies by learning from natural antibody-antigen complexes Luo et al. (2022); Jin et al. (2022); Kong et al. (2023a;b); Bennett et al. (2024). Despite significant strides in protein design Dauparas et al. (2022); Hsu et al. (2022); Notin et al. (2024), antibodies present a distinct challenge for deep learning due to the high flexibility of their binding regions, known as complementarity-determining regions (CDRs). Inspired by RFdiffusion's Watson et al. (2023) remarkable achievements in monomeric protein and binder design, Bennett et al. Bennett et al. (2024) advanced the field by fine-tuning the

[*]Equal contribution
[†]Work done when doing an internship at Tencent
[‡]Corresponding authors: zkf1105@gmail.com, brunojiang@tencent.com

RFdiffusion model with antibody-antigen complex structural data to facilitate epitope-targeted antibody design. Their approach aligns well with established pharmaceutical practices by generating different CDRs on the same framework for different antigen targets, thereby enhancing developability and reducing downstream optimization requirements. While structural and functional analyses validated the its capability to generate antibodies that bind to predetermined epitopes, the approach was constrained by notably low success rates.

One reason for the low success rate of this AI-based antibody design pipeline is the occurrence of hallucinations, especially during the process of protein inverse folding, which predicts the CDR sequence based on the backbone structure Dauparas et al. (2022); Hsu et al. (2022); Gruver et al. (2023); Gao et al. (2023b). To be specific, given an antigen epitope and an antibody backbone, the amino acid sequences inferred through methods like ProteinMPNN Dauparas et al. (2022) and ESM-IF1 Hsu et al. (2022) may not fold into the desired structures in real biological systems. More critically, there are currently no effective computational methods to reduce these hallucinations, aside from conducting time-consuming, labor-intensive, and expensive wet-lab experiments for validation. Typically, using independent structure prediction models to fold and verify the inferred sequences cannot effectively eliminate non-functional sequences caused by hallucinations. That is because even state-of-the-art models exhibit structural deviations of 1 to 3 Å and have low confidence in predicting the structures of antibody CDRs.

To deal with the challenge of hallucinations arising from previous models, we propose an antibody CDR sequence design framework from a novel perspective of similar structure retrieval, named as IgSeek (Ig for Immunoglobulin, a.k.a. antibody). Our framework is enlightened by a noteworthy empirical discovery made 25 years ago, which revealed that antibodies exhibit a limited set of canonical structures within 5 out of 6 CDRs despite the vast diversity in sequences, and that certain CDR conformations are scaffolded by a few highly conserved residues Chothia et al. (1989). Further inspired by retrieval-augmented prediction for hallucination reduction in protein structure prediction Jumper et al. (2021), and natural language generation Gao et al. (2023a), IgSeek leverages neural retrieval in a database of natural antibodies to retrieve structurally similar sequence templates of CDR, and ensembles the queried templates for sequence prediction. Extensive experimental validation demonstrates that our structure-guided retrieval approach effectively improves the accuracy of CDR sequence prediction, notably outperforming state-of-the-art sequence design methods.

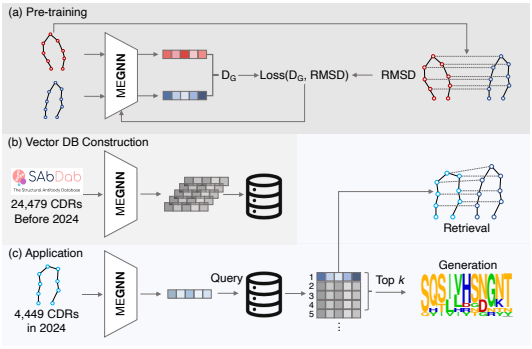

Figure 1: The Framework IgSeek: (a) Pre-train an MEGNN encoder by a self-supervised learning task. (b) Construct a CDR vector database. (c) Sequence generation by K-NN search.

2 Results

2.1 IgSeek Approach

The gist of IgSeek for structure-to-sequence generation is isomorphic structure retrieval, which allows for the exploration of a large and diverse antibody CDR structure database. Fig. 1 illustrates the framework of IgSeek. Given an antibody CDR database where both structures and sequences are available, IgSeek first constructs a CDR vector database, where vector embeddings index the structural proximity of the CDRs. In this offline stage, we pre-train a *Multi-channel Equivariant Graph Neural Network (MEGNN)* to encode the structure of CDR loops into fixed-length vectors within the CDR database. Specifically, MEGNN aligns the spatial structure distance between pairs of CDRs with equal lengths and similar conformations. Subsequently, for a CDR structure G whose sequence is to be predicted, we first deploy the pre-trained MEGNN to generate an embedding $\boldsymbol{h}_G$ for G. $\boldsymbol{h}_G$ then serves as the search key to query the *K-nearest neighbors (K-NN)* structurally similar CDR loops in the vector database. Finally, the K-NN results, associated with their corresponding

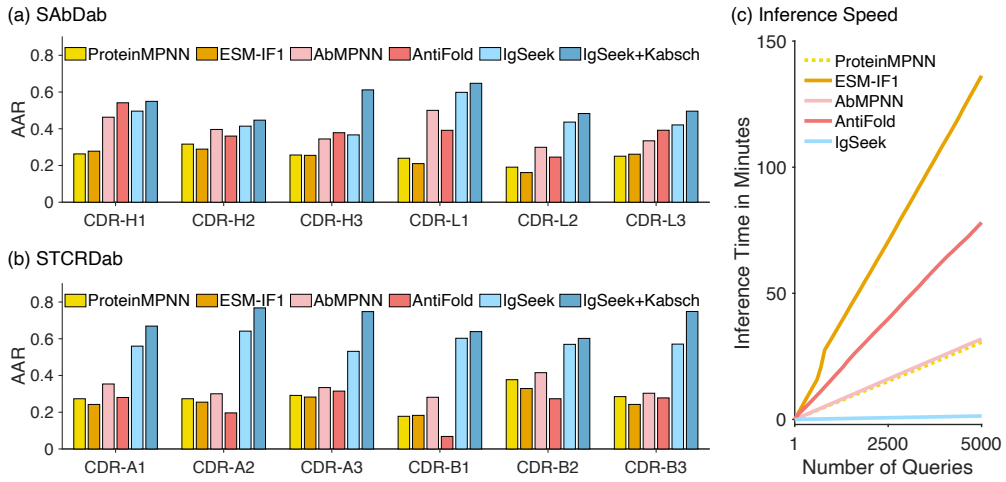

Figure 2: The comparison of average AAR and inference speed. (a) AAR in SAbDab-2024 dataset. (b) AAR in STCRDab. (c) Inference speed.

residue sequences, are collected to predict the sequence of G by ensemble and Bernoulli sampling.A detailed description of our methodology can be found in Appendix C.

Datasets. We evaluate our IgSeek and other baselines using both solved and predicted antibody structures. The training set consists of CDR pairs sampled from $11,023$ solved CDR loops in the *Structural Antibody Database (SAbDab)* (Dunbar et al., 2013; Schneider et al., 2021). To construct the CDR vector database, we utilize $24,479$ solved CDR loops from SAbDab before January 1, 2024 (SAbDab-before-2024). In addition, $4,449$ solved CDR loops released between January 3, 2024 and May 29, 2024 from SAbDab (SAbDab-2024) serve as the test set to evaluate the performance of IgSeek and its competitors. In addition to the solved antibody structures from SAbDab, we also conduct experiments on $5,111$ CDR loops from the *Structural T-Cell Receptor Database (STCRDab)* (Leem et al., 2018) to evaluate the model generalization ability. Furthermore, we evaluate the model efficiency using $5,000$ predicted CDR-H3 loops from the *Observed Antibody Space (OAS-H3)* (Kovaltsuk et al., 2018; Olsen et al., 2022). IMGT numbering scheme Lefranc et al. (2003) is utilized for antibody datasets. More details of datasets and experiment settings can be found in Appendix D and Appendix E.

2.2 IgSeek for CDR Structure Retrieval

In this set of experiments, we compare IgSeek with the state-of-the-art structure searching model, FoldSeek Van Kempen et al. (2024), by examining the quality of the retrieved isomorphic structures. Introduction to competitors is deferred to Appendix F. Specifically, for a given query CDR q, the retrieved CDR r is considered a positive instance if their RMSD is less than 1 Å. To ensure the robustness of our evaluation, we omit any query CDR for which there are no candidates in the CDR database with a distance of less than 1 Å from the query. This strategy allows us to focus on instances where meaningful comparisons can be made, thereby enhancing the result reliability.

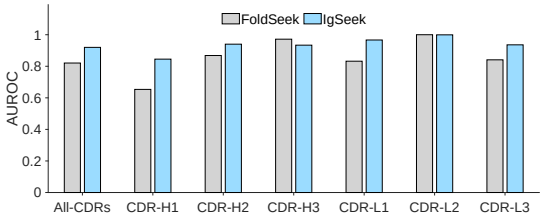

Figure 3: IgSeek vs. FoldSeek in CDR retrieval.

Fig. 3 presents the experimental results of IgSeek and FoldSeek, illustrating the model performance on the retrieved sequences using the AUROC metric. As we can observe, IgSeek outperforms FoldSeek on four types of CDR loops while maintaining comparable performance on CDR-H3 and CDR-L1, indicating its capability of identifying structurally similar CDRs across diverse CDR loops. It is worth noting that IgSeek achieves a 2.6x speed-up in structure retrieval time compared to FoldSeek. Since this improvement in speed does not come at the cost of accuracy, it demonstrates that IgSeek

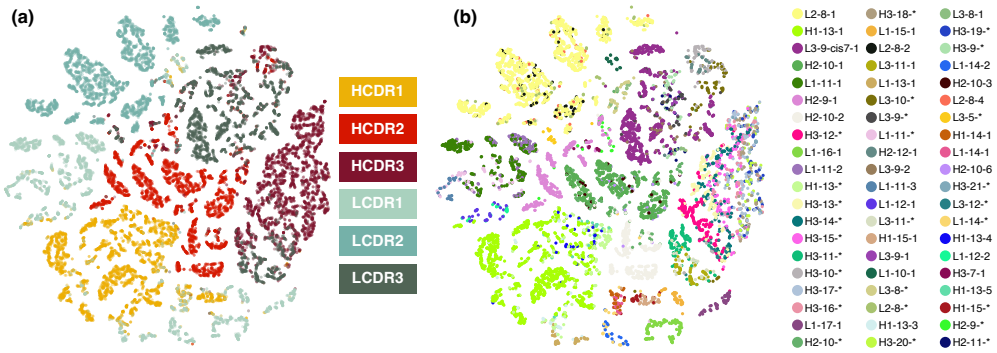

Figure 4: Embeddings of CDRs in the SAbDab-before-2024 datasets projected onto 2D Space.

strikes a superior trade-off between efficiency and accuracy. The ability to quickly retrieve high-quality structural matches can greatly enhance workflows in antibody design, as shown in Sec. 2.3.

2.3 IgSeek for CDR Sequence Design

In this set of experiments, we compare IgSeek with the state-of-the-art models for protein and antibody sequence design, including ProteinMPNN (Dauparas et al., 2022), ESM-IF1 (Hsu et al., 2022), AbMPNN (Dreyer et al., 2023), and AntiFold (Høie et al., 2024). Specifically, the MEGNN in IgSeek is trained on the SAbDab-before-2024 dataset to construct the CDR vector database. Subsequently, the trained MEGNN is utilized to generate embeddings for the CDRs in the SAbDab-2024 dataset. For each query CDR in the SAbDab-2024 dataset, we retrieve the top-10 nearest neighbors from the SAbDab-before-2024 dataset in the CDR vector database, ensuring that the lengths of the retrieved sequences match that of the query. Finally, we proceed to sample the amino acids for each position in the CDR sequences to generate the predicted result for the query CDR. Note that existing protein and antibody inverse folding methodologies such as ProteinMPNN and AntiFold typically generate at least two samples for evaluation. In our experiments, we follow the settings of ProteinMPNN and present the best results of all other methods for evaluation. Average *amino acid recovery (AAR)* is utilized to evaluate model performance, which quantifies the accuracy of the predicted sequences. For a query CDR q, the AAR is defined as the ratio of overlapping positions between the predicted sequence $\hat{\boldsymbol{s}}_q$ and ground-truth sequence $\boldsymbol{s}_q$: $\text{AAR}\left(\hat{\boldsymbol{s}}_q, \boldsymbol{s}_q\right) = \frac{1}{L}\sum_{l=1}^{L}\mathbb{I}(\hat{\boldsymbol{s}}_q(l), \boldsymbol{s}_q(l))$.

Fig. 2 (a) illustrates the average AAR for each model on the SAbDab-2024 dataset. As we can observe, Antifold and AbMPNN achieve much better results compared to ProteinMPNN and ESM-IF1, highlighting the advantages of fine-tuning pre-trained protein design models specifically on the antibody dataset. Additionally, IgSeek outperforms its competitors by at least 2.9% on light chain CDR loops (CDR-L) and achieves results comparable to state-of-the-art methods on heavy chain CDR loops (CDR-H). We incorporate an additional variant of IgSeek that uses RMSD as a secondary sorting metric, denoted as IgSeek+Kabsch. Notice that we do not deploy the Kabsch algorithm to search the entire database. Instead, we validated the RMSD of the top-ranked CDRs identified by IgSeek until we identified the top 10 CDRs with RMSD less than 1 Å. Notably, IgSeek+Kabsch consistently outperforms all baselines across six types of CDR loops, highlighting the effectiveness of our retrieval-based strategy. The marked advantage of IgSeek+Kabsch on CDR-H3 loops is particularly noteworthy, as CDR-H3 is often considered one of the most hypervariable regions.

Remark. We observe a performance degradation in AntiFold and AbMPNN on the SAbDab-2024 dataset compared to the results reported by Høie et al. (2024). One possible reason for this discrepancy is that these two models heavily depend on antibody backbone structures as auxiliary information, while only the structures of CDRs are given in our settings.

Generalization Performance. Next, we evaluate the model inference performance on the STCRDab dataset without any further model training. To conduct this evaluation, we randomly draw around 80% of the CDR loops to generate selection templates, while the remaining 20% are used as queries. Fig. 2 (b) displays the average AAR of each model on the STCRDab dataset. As we can see, IgSeek takes the lead by at least 30% on CDR loops from chain A and chain B, respectively. These impressive results further underscore the potential of structure retrieval approaches in mitigat-

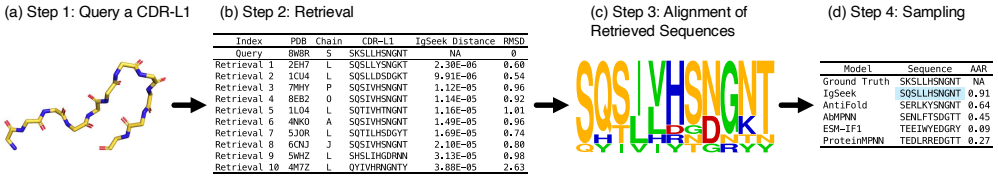

Figure 5: A Case study using 8W8R CDR-L1 as an example.

Table 1: Igseek with structure generators on 8R1C CDR-L1.

Model	Sequence	AAR↑	RMSD↓
Ground Truth	SSDVGSYNL	-	-
dyMEAN	SSQSLLYSS	0.33	4.10
dyMEAN⇒Igseek	SSNIGSGYD	0.44	4.10
RFdiffusion⇒Igseek	SSDIGAYND	0.67	0.38

ing hallucinations during sequence inference, demonstrating that IgSeek can effectively generalize to unseen data while maintaining high accuracy in sequence recovery.

Efficiency Evaluation. We evaluate the model efficiency using the OAS-H3 dataset. Fig. 2 (c) reports the inference time of IgSeek compared with other baseline models, all without any model retraining. As we can observe, IgSeek achieves at least 20x speed-up compared to baseline methods, which demonstrates that our IgSeek achieves a better trade-off between effectiveness and efficiency. This enhanced inference speed is particularly beneficial in practical applications like high-throughput antibody design where rapid sequence generation is crucial.

Visualization. To investigate the representation generated by MEGNN, we conduct a visualization analysis on the SAbDab-before-2024 dataset by T-SNE (Van der Maaten & Hinton, 2008). Fig. 4 presents the visualization results of top-60 CDR representations in each cluster, where PyIgClassify cluster labels (Adolf-Bryfogle et al., 2015) (refer to Appendix D) are utilized in this set of experiments. As Fig. 4 illustrates, IgSeek produces a high-quality visualization that clearly organizes the embeddings of CDR loops from distinct clusters into separate groups with minimal overlaps. Furthermore, the visualization not only demonstrates the effectiveness of IgSeek in distinguishing CDRs among different clusters but also highlights its ability to capture structural information inherent in CDR loops. This visual clarity and distinct grouping underscore the robustness and discriminative capability of IgSeek in embedding isomorphic CDR structures closer together while ensuring distinct clusters remain well-separated, which facilitates the identification and retrieval of CDR loops based on their structural characteristics.

2.4 Case Study

Example. We first use the 8W8R CDR-L1 as an example to illustrate the query and generation process of IgSeek. Step 1: given the backbone structure of the 8W8R CDR-L1 loop, we employ the pre-trained MEGNN to generate its embeddings. Step 2: we retrieve the top-10 nearest neighbors of the 8W8R CDR-L1 loop from the CDR vector database $\mathcal{Z}$. Step 3: we utilize the aligned sequences from the retrieved records to generate the residue probability distribution at each position. Step 4: Finally, we sample the output result from this distribution. In this example, we observe that the AAR of the sequence generated by IgSeek outperforms other competitors by at least 0.27, demonstrating the effectiveness of our approach.

Incorporation with structure generation models. Next, we evaluate the sequence prediction capabilities of Igseek using dyMEAN Kong et al. (2023a) and RFdiffusion Watson et al. (2023) as structure generators for unseen antibody structures, focusing on 8R1C CDR-L1 loop. The pipeline first employs dyMEAN or RFdiffusion to generate missing CDR structures, followed by sequence predictions by Igseek for this loop, and the experimental results are presented in Table 1.The analysis reveals significant improvement in sequence recovery rates for 8SGN CDR-L1 when using Igseek, demonstrating the effectiveness of Igseek as a key component in real-world antibody design pipelines. In particular, our experiments indicate that the quality of structural prediction substantially influences the accuracy of sequence generation, suggesting that the development of precise CDR structure prediction models would further enhance the overall antibody design pipeline.

3 CONCLUSION

In this paper, we propose an antibody sequence design framework, IgSeek, from a new learning-based structure retrieval perspective. Specifically, IgSeek first constructs a CDR vector database using a multi-channel equivariant graph neural network. It then predicts CDR sequences from templates retrieved from isomorphic structures in the database. Extensive experiments demonstrate the effectiveness and efficiency of IgSeek, providing insights into de novo antibody sequence design and can inspire further investigation in this direction.

ACKNOWLEDGMENTS

Sibo Wang is supported by the RGC GRF grant (No. 14217322), Hong Kong ITC ITF grant (No. MRP/071/20X), and Tencent Rhino-Bird Focused Research Grant. Kangfei Zhao is supported by National Key Research and Development Plan, No. 2023YFF0725101.

REFERENCES

Jonathan Abraham. Passive antibody therapy in covid-19. *Nature Reviews Immunology*, 20(7): 401–403, 2020.

Gregory P Adams and Louis M Weiner. Monoclonal antibody therapy of cancer. *Nature biotechnology*, 23(9):1147–1157, 2005.

Jared Adolf-Bryfogle, Qifang Xu, Benjamin North, Andreas Lehmann, and Roland L Dunbrack Jr. Pyigclassify: a database of antibody cdr structural classifications. *Nucleic acids research*, 43(D1): D432–D438, 2015.

Jared Adolf-Bryfogle, Oleks Kalyuzhniy, Michael Kubitz, Brian D Weitzner, Xiaozhen Hu, Yumiko Adachi, William R Schief, and Roland L Dunbrack Jr. Rosettaantibodydesign (rabd): A general framework for computational antibody design. *PLoS computational biology*, 14(4):1–38, 2018.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, Claudia Millán, Hahnbeom Park, Carson Adams, Caleb R Glassman, Andy DeGiovanni, Jose H Pereira, Andria V Rodrigues, Alberdina A Van Dijk, Ana C Ebrecht, Diederik J Opperman, Theo Sagmeister, Christoph Buhlheller, Tea Pavkov-Keller, Manoj K Rathinaswamy, Udit Dalwadi, Calvin K Yip, John E Burke, K Christopher Garcia, Nick V Grishin, Paul D Adams, Randy J Read, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):2453, 2022.

Nathaniel R Bennett, Joseph L Watson, Robert J Ragotte, Andrew J Borst, Déjenaé L See, Connor Weidle, Riti Biswas, Ellen L Shrock, Philip JY Leung, Buwei Huang, Inna Goreshnik, Russell Ault, Kenneth D Carr, Benedikt Singer, Cameron Criswell, Dionne Vafeados, Mariana Garcia Sanchez, Ho Min Kim, Susana Vázquez Torres, Sidney Chan, and David and Baker. Atomically accurate de novo design of single-domain antibodies. *bioRxiv*, 2024.

Ginger Chao, Wai L Lau, Benjamin J Hackel, Stephen L Sazinsky, Shaun M Lippow, and K Dane Wittrup. Isolating and engineering human antibodies using yeast surface display. *Nature protocols*, 1(2):755–768, 2006.

Cyrus Chothia, Arthur M. Lesk, Anna Tramontano, Michael Levitf, Sandra J. Smith-GiII, Gillian Air, Steven Sheriff, Eduardo A. Padlan, David Davies, William R. Tulip, Peter M. Colman, Silvia Spinelli, Pedro M. Alzari, and Roberto J. Poljak. Conformations of immunoglobulin hypervariable regions. *Nature*, 342(6252):877–883, 1989.

George V. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Frédéric A Dreyer, Daniel Cutting, Constantin Schneider, Henry Kenlay, and Charlotte M Deane. Inverse folding for antibody sequence design using deep learning. In *ICML CompBio*, 2023.

James Dunbar, Konrad Krawczyk, Jinwoo Leem, Terry Baker, Angelika Fuchs, Guy Georges, Jiye Shi, and Charlotte M. Deane. Sabdab: the structural antibody database. *Nucleic Acids Res.*, 42 (D1):D1140–D1146, 2013.

Kevin Dunleavy. Who's no. 1? with $25b in sales, merck's keytruda looks to be the top-selling drug of 2023. *Fierce Pharma*, 2024.

Marc Feldmann and Ravinder N Maini. Tnf defined as a therapeutic target for rheumatoid arthritis and other autoimmune diseases. *Nature medicine*, 9(10):1245–1250, 2003.

Brendan J Frey and Delbert Dueck. Clustering by passing messages between data points. *science*, 315(5814):972–976, 2007.

Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se(3)-transformers: 3d roto-translation equivariant attention networks. In *NeurIPS*, pp. 1970–1981, 2020.

Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3):183–192, 1989.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *CoRR*, abs/2312.10997, 2023a.

Zhangyang Gao, Cheng Tan, Pablo Chacón, and Stan Z Li. Pifold: Toward effective and efficient protein inverse folding. In *ICLR*, 2023b.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim G. J. Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. In *NeurIPS*, volume 36, pp. 12489–12517, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

Magnus Høie, Alissa Hummer, Tobias Olsen, Morten Nielsen, and Charlotte Deane. Antifold: Improved antibody structure design using inverse folding. *arXiv*, 2024. URL `https://arxiv.org/abs/2405.03370`.

Liisa Holm. Using dali for protein structure comparison. *Structural Bioinformatics: Methods and Protocols*, pp. 29–42, 2020.

Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4 (2):251–257, 1991.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. In *ICML*, pp. 8946–8970, 2022.

Wenbing Huang, Jiaqi Han, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. Equivariant graph mechanics networks with constraints. In *ICLR*, 2022.

Wengong Jin, Jeremy Wohlwend, Regina Barzilay, and Tommi Jaakkola. Iterative refinement graph neural network for antibody sequence-structure co-design. In *ICLR*, 2022.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations*, 2021.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon AA Kohl, Andrew J Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Diederik P Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. End-to-end full-atom antibody design. In *ICML*, pp. 17409–17429, 2023a.

Xiangzhe Kong, Wenbing Huang, and Yang Liu. Conditional antibody design as 3d equivariant graph translation. In *ICLR*, 2023b.

Aleksandr Kovaltsuk, Jinwoo Leem, Sebastian Kelm, James Snowden, Charlotte M Deane, and Konrad Krawczyk. Observed antibody space: a resource for data mining next-generation sequencing of antibody repertoires. *J. Immunol.*, 201(8):2502–2509, 2018.

Jinwoo Leem, Saulo H P de Oliveira, Konrad Krawczyk, and Charlotte M Deane. Stcrdab: the structural t-cell receptor database. *Nucleic acids research*, 46(D1):D406–D412, 2018.

Marie-Paule Lefranc, Christelle Pommié, Manuel Ruiz, Véronique Giudicelli, Elodie Foulquier, Lisa Truong, Valérie Thouvenin-Contet, and Gérard Lefranc. Imgt unique numbering for immunoglobulin and t cell receptor variable domains and ig superfamily v-like domains. *Developmental & Comparative Immunology*, 27(1):55–77, 2003.

Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *ICLR*, 2023.

Shitong Luo, Yufeng Su, Xingang Peng, Sheng Wang, Jian Peng, and Jianzhu Ma. Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures. *NeurIPS*, 35:9754–9767, 2022.

Robert M MacCallum, Andrew CR Martin, and Janet M Thornton. Antibody-antigen interactions: contact analysis and binding site topography. *Journal of molecular biology*, 262(5):732–745, 1996.

Benjamin North, Andreas Lehmann, and Roland L Dunbrack Jr. A new clustering of antibody cdr loop conformations. *Journal of molecular biology*, 406(2):228–256, 2011.

Pascal Notin, Nathan Rollins, Yarin Gal, Chris Sander, and Debora Marks. Machine learning for functional protein design. *Nature biotechnology*, 42(2):216–228, 2024.

Tobias H Olsen, Fergus Boyles, and Charlotte M Deane. Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. *Protein Science*, 31(1):141–146, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pp. 8024–8035, 2019.

David Procházka, Terézia Slanináková, Jaroslav Olha, Adrián Rošinec, Katarína Grešová, Miriama Jánošová, Jakub Čillík, Jana Porubská, Radka Svobodová, Vlastislav Dohnal, and Matej Antol. Alphafind: discover structure similarity across the proteome in alphafold db. *Nucleic Acids Research*, 52(W1):W182–W186, 2024.

Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *ICML*, pp. 9323–9332, 2021.

Constantin Schneider, Matthew I J Raybould, and Charlotte M Deane. Sabdab in the age of biotherapeutics: updates including sabdab-nano, the nanobody structure tracker. *Nucleic Acids Res.*, 50 (D1):D1368–D1372, 2021.

Ilya N Shindyalov and Philip E Bourne. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein engineering*, 11(9):739–747, 1998.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958, 2014.

Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, pp. 6306–6315, 2017.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(11), 2008.

Michel Van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nature biotechnology*, 42(2):243–246, 2024.

Jean P Van Wauwe, JR De Mey, and JG Goossens. Okt3: a monoclonal anti-human t lymphocyte antibody with potent mitogenic properties. *Journal of immunology (Baltimore, Md.: 1950)*, 124 (6):2708–2713, 1980.

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, Augustin Žídek, Tim Green, Kathryn Tunyasuvunakool, Stig Petersen, John Jumper, Ellen Clancy, Richard Green, Ankur Vora, Mira Lutfi, Michael Figurnov, Andrew Cowie, Nicole Hobbs, Pushmeet Kohli, Gerard Kleywegt, Ewan Birney, Demis Hassabis, and Sameer Velankar. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic acids research*, 50(D1):D439–D444, 2022.

Rubo Wang, Fandi Wu, Xingyu Gao, Jiaxiang Wu, Peilin Zhao, and Jianhua Yao. Iggm: A generative model for functional antibody and nanobody design. *bioRxiv*, 2024.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, Basile I M Wicky, Nikita Hanikel, Samuel J Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with rfdiffusion. *Nature*, 620 (7976):1089–1100, 2023.

Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical evaluation of rectified activations in convolutional network. *arXiv*, 2015. URL `https://arxiv.org/abs/1505.00853`.

Chengxin Zhang, Morgan Shine, Anna Marie Pyle, and Yang Zhang. Us-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes. *Nature methods*, 19(9): 1109–1115, 2022.

Yang Zhang and Jeffrey Skolnick. Tm-align: a protein structure alignment algorithm based on the tm-score. *Nucleic acids research*, 33(7):2302–2309, 2005.

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein representation learning by geometric structure pretraining. In *ICLR*, 2023.

APPENDIX

A RELATED WORK

Protein Structure Retrieval. With the growth of the volume of protein structures, structure retrieval has become a critical task in protein data management. AlphaFind (Procházka et al., 2024) is a web

Table 2: Settings of different antibody (Ab) design tasks.

Category	Input			Output	
	Ab Framework	Ab CDR	Antigen	CDR Structure	CDR Sequence
Antibody Inverse Folding	✓	✓	✗	✗	✓
Antibody Co-design	✓	✓	✓	✓	✓
Sequence Design (ours)	✗	✓	✗	✗	✓

tool designed to identify structurally similar proteins in AlphaFold Database (Varadi et al., 2022) by compressing data from $\sim$23 TB to $\sim$20 GB using vector embeddings, narrowing down candidates with a neural network. The similarity of the search result is evaluated by US-align (Zhang et al., 2022). Another state-of-the-art method, FoldSeek (Van Kempen et al., 2024), accelerates protein structure searches by representing tertiary amino acid interactions as sequences over a 3D interaction structural alphabet, which derives from vector quantization by VQ-VAE (van den Oord et al., 2017). However, the representation only models the structure of two contiguous residues in a chain.

Protein Inverse Folding. Protein inverse folding aims to predict diverse sequences that can fold into a given protein structure. ProteinMPNN (Dauparas et al., 2022) is a deep learning–based method for protein sequence design that excels in both in silico and experimental evaluations. By leveraging a message-passing neural network with enhanced input features and edge updates, ProteinMPNN is capable of designing monomers, cyclic oligomers, protein nanoparticles, and protein-protein interfaces, rescuing previously failed designs generated by Rosetta (Adolf-Bryfogle et al., 2018; Baek et al., 2021) or AlphaFold (Jumper et al., 2021). ESM-IF1 (Hsu et al., 2022) employs a sequence-to-sequence Transformer to predict protein sequences from backbone atom coordinates.

Antibody Inverse Folding. AbMPNN (Dreyer et al., 2023) inherits the model architecture of ProteinMPNN, and trains an antibody-specific variant for antibody design. It outperforms generic protein design models in sequence recovery and structure robustness, especially for hyper-variable CDR-H3 loops. AntiFold (Høie et al., 2024) is an antibody-specific inverse folding model, which is fine-tuned on ESM-IF1, with both solved and predicted antibody structures. However, it should be emphasized that Antifold infers CDR sequences based on the structure of the variable domain and the sequence of the framework regions. Consequently, the accuracy of CDR sequence inference is influenced not only by the structure of the CDRs but also by the sequence and structural information of the framework regions. Previous studies utilizing antibody sequence language models without structural information have demonstrated that the sequence of the framework regions can partially predict the CDR sequences, particularly for relatively conserved residues. As a result, the requirement for the framework sequence as input complicates the inference of CDR sequences that can bind to different antigens while maintaining an identical framework.

Antibody Co-Design. In recent years, deep learning models have emerged as powerful data-driven approaches for antibody design. RefineGNN (Jin et al., 2022) is the first structure sequence co-design method that alternatively predicts the atom coordinates and residue types in CDRs by auto-regression. DiffAb (Luo et al., 2022) and IgGM (Wang et al., 2024) utilize diffusion models to generate the structure and sequence of CDRs based on the framework regions and the target antigen, with DiffAb oriented for specific antigens. MEAN (Kong et al., 2023b) and dyMEAN (Kong et al., 2023a) employ graph neural networks to predict the structure and sequence of CDRs. Table 2 presents a comparative analysis of various antibody design task configurations.

B Preliminaries and Problem Formulation

Antibodies are sophisticated Y-shaped protein characterized by their distinctive structural architecture, comprising two identical sets of polypeptide chains. Each set consists of a heavy chain and a light chain, with both chains exhibiting a modular organization of constant and variable regions. While the constant regions maintain high sequence conservation across different antibody molecules, the variable regions display significant diversity, enabling specific antigen recognition and binding capabilities. Within the variable domains, the structural organization follows a precise pattern defined by the IMGT numbering scheme, alternating between four framework regions (FRs) and three complementarity determining regions (CDRs). The FRs provide the structural scaffold,

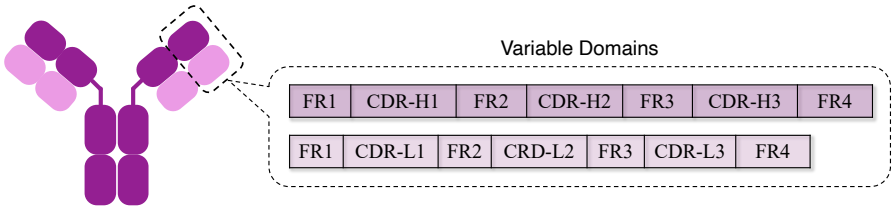

Figure 6: Antibody Structure

while the CDRs form the antigen-binding interface and are primarily responsible for the specificity and affinity of antigen recognition. These hypervariable CDR loops represent the most critical determinants of antibody-antigen interactions and are consequently the primary focus of rational antibody engineering and design efforts.

We represent the 3D structure of a CDR as a geometric graph $G = (V, E)$ with node set V and edge set E (Jing et al., 2021; Jin et al., 2022; Zhang et al., 2023). Each node $v_i \in V$ denotes an amino acid residue, associated with a multi-channel 3D coordinate matrix $\boldsymbol{X}_i \in \mathbb{R}^{c\times 3}$, where c is the channel size, i.e., the number of atoms in the residue v_i. In this paper, we consider the four backbone atoms $\{\text{N}, \text{C}_\alpha, \text{C}, \text{O}\}$ that are independent to residue type, i.e., $c = 4$. Each edge $e_{ij} \in E$ denotes an interaction between v_i and v_j, if the Euclidean distance between their C_α atoms is within a threshold θ. The neighborhood of a node v_i, denoted as $\mathcal{N}_i$, consists of the adjacency nodes of v_i, that is, $\{v_j|(v_i, v_j) \in E\}$.

CDR Sequence Design. Given the structure $G = (E, V)$ of a CDR and the multi-channel 3D coordinate of each residue, in this paper, we aim to reconstruct the corresponding sequence of the CDR, denoted as $\boldsymbol{s} = \{s(i)|i \in [1, \cdots, |V|]\}$, where $s(i)$ is the amino acid type of residue v_i.

E(3) Equivalence is an important property in modeling the 3D structures (Fuchs et al., 2020; Batzner et al., 2022; Liao & Smidt, 2023). Formally, let $\mathcal{X}$ and $\mathcal{Y}$ be two vector spaces, with $T_{\mathcal{X}}(g) : \mathcal{X} \to \mathcal{X}$ and $T_{\mathcal{Y}}(g) : \mathcal{Y} \to \mathcal{Y}$ representing two sets of transformations for the abstract group $g \in E(3)$. A function $\phi : \mathcal{X} \to \mathcal{Y}$ is E(3) Equivariant to g if it satisfies the following condition:

$$\phi(\{T_{\mathcal{X}}(g)\boldsymbol{x}_i, \boldsymbol{h}_i\}_{i=1}^n) = T_{\mathcal{Y}}(g)\phi(\{\boldsymbol{x}_i, \boldsymbol{h}_i\}_{i=1}^n), \tag{1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^3$ denotes the input 3D coordinates and $\boldsymbol{h}_i \in \mathbb{R}^d$ is the d-dimensional features of a node, respectively. This inductive bias guarantees that ϕ preserves equivariant transformation regarding transformation of the coordinate system in E(3) group (Satorras et al., 2021; Huang et al., 2022; Liao & Smidt, 2023). A typical example for this transformation operation in the space $\mathcal{X}$ is given by $T_{\mathcal{X}}(g)\boldsymbol{x}_i^{(0)} = \boldsymbol{R}\boldsymbol{x}_i^{(0)} + \boldsymbol{b}$, where $\boldsymbol{R} \in \mathbb{R}^{3\times 3}$ is an orthogonal matrix and $\boldsymbol{b}$ is the bias term.

To achieve equivalence, equivariant graph neural networks are proposed (Satorras et al., 2021; Huang et al., 2022; Kong et al., 2023a;b), which follows a general message-passing framework as shown in Eq. 2-4. Here, $\boldsymbol{m}_{j\to i}^{(l)}$ denotes the messages propagated from node v_j to v_i, and $d_{ij}^{(l-1)} = dist(v_i, v_j)$ denotes the Euclidean distance between v_i and v_j, and $\boldsymbol{x}_{ij}^{(l-1)}$ denotes coordinate differences between v_i and v_j at the $(l-1)$-th layer.

$$\boldsymbol{m}_{j\to i}^{(l)} = \psi_1\left(\boldsymbol{h}_i^{(l-1)}, \boldsymbol{h}_j^{(l-1)}, \boldsymbol{x}_{ij}^{(l-1)}, d_{ij}^{(l-1)}\right), \tag{2}$$

$$\boldsymbol{h}_i^{(l)} = \psi_2\left(\boldsymbol{h}_i^{(l-1)}, \sum_{v_j\in\mathcal{N}_i} \boldsymbol{m}_{j\to i}^{(l)}\right), \tag{3}$$

$$\boldsymbol{x}_i^{(l)} = \psi_3\left(\boldsymbol{x}_i^{(l-1)}, \boldsymbol{x}_{ij}^{(l-1)} \sum_j \psi_4\left(\sum_{v_j\in\mathcal{N}_i} \boldsymbol{m}_{j\to i}^{(l)}\right)\right). \tag{4}$$

The functions $\{\psi_1, \psi_2, \psi_3, \psi_4\}$ are equivariant transformations, typically implemented as Multi-Layer Perceptrons (MLPs) to leverage the universal approximation (Funahashi, 1989; Cybenko, 1989; Hornik, 1991). In this process, the feature $\boldsymbol{h}_i^{(l)}$ remains E(3) invariant, while the coordinate $\boldsymbol{x}_i^{(l)}$ is E(3) equivariant.

C METHODOLOGY: IGSEEK

In this section, We present IgSeek, a retrieval-based framework for CDR sequence design. The core of IgSeek is isomorphic structure retrieval from a comprehensive antibody CDR database. As shown in Fig. 1, IgSeek first builds a vector database using a pre-trained Multi-channel Equivariant Graph Neural Network (MEGNN) to encode CDR structural information. For a target CDR structure G, IgSeek generates its embedding using MEGNN, retrieves K-nearest structurally similar CDR loops from the database, and predicts G's sequence through ensemble and Bernoulli sampling of the retrieved sequences. In the following, we will present the model design of the MEGNN encoder in Section C.1, discuss the learning objective and the sequence prediction in Section C.2, followed by model analysis in Section C.3.

C.1 MULTI CHANNEL EQUIVARIANT ENCODER

Recall that each amino acid residue v_i is represented by its four backbone atoms, thereby we extend the general single-channel EGNN layer (Satorras et al., 2021; Huang et al., 2022) to a multi-channel layer, with each channel corresponding to a specific atom. Unlike existing approaches (Kong et al., 2023a; Høie et al., 2024) that leverage domain knowledge of the well-conserved antibody backbone structure, our MEGNN encoder generates CDR embeddings exclusively based on the antibody CDR structure, without relying on any prior backbone knowledge.

For a 3D CDR structure G, the MEGNN encoder takes the initial features of each residue v_i, denoted as $\boldsymbol{h}_i^{(0)} \in \mathbb{R}^d$, along with the perturbed coordinates $\hat{\boldsymbol{X}}_i \in \mathbb{R}^{c\times 3}$ as input. Here, c denotes the number of atoms, which is set to $c = 4$, and $\boldsymbol{h}_i^{(0)}$ is initialized by a uniform distribution. $\hat{\boldsymbol{X}}_i = \boldsymbol{X}_i + \mathcal{N}(0, \sigma)$, where $\mathcal{N}(0, \sigma)$ denotes a small Gaussian noise. This perturbation introduces variability that enhances the robustness of the model.

Multi-channel Equivariant Message Passing. The l-th layer of MEGNN updates both the node features $\boldsymbol{h}_i^{(l)}$ and coordinates $\boldsymbol{X}_i^{(l)}$ by Eq. 5-8, where ρ is a distance computation function, ϕ_e, ϕ_X and ϕ_h are neural network transformations. The update process is defined as follows:

$$\boldsymbol{X}_{ij}^{(l-1)}, \boldsymbol{z}_{ij}^{(l-1)} = \rho\left(\boldsymbol{X}_i^{(l-1)}, \boldsymbol{X}_j^{(l-1)}, e_{ij}\right), \tag{5}$$

$$\boldsymbol{h}_{e_{ij}}^{(l)} = \phi_e\left(\text{CONCAT}\left(\boldsymbol{h}_i^{(l-1)}, \boldsymbol{h}_j^{(l-1)}, \boldsymbol{z}_{ij}^{(l-1)}\right)\right), \tag{6}$$

$$\boldsymbol{X}_i^{(l)} = \phi_X\left(\boldsymbol{X}_i^{(l-1)}, \{\boldsymbol{h}_{e_{ij}}^{(l)}, \boldsymbol{X}_{ij}^{(l-1)} | v_j \in \mathcal{N}_i\}\right), \tag{7}$$

$$\boldsymbol{h}_i^{(l)} = \phi_h\left(\boldsymbol{h}_i^{(l-1)}, \{\boldsymbol{h}_{e_{ij}}^{(l)} | v_j \in \mathcal{N}_i\}\right). \tag{8}$$

Specifically, MEGNN first computes the coordinate differences $\boldsymbol{X}_{ij}^{(l-1)}$ and the square distance $\boldsymbol{z}_{ij}^{(l-1)}$ between each pair of backbone atoms among different residues in ρ (Eq. 5) as below:

$$\boldsymbol{X}_{ij}^{(l-1)} = \boldsymbol{X}_i^{(l-1)} - \boldsymbol{X}_j^{(l-1)}, \quad \boldsymbol{z}_{ij}^{(l-1)} = (\boldsymbol{X}_{ij}^{(l-1)})^\top \boldsymbol{X}_{ij}^{(l-1)}.$$

Subsequently, an edge module ϕ_e generates the edge feature $\boldsymbol{h}_{e_{ij}}^{(l)}$ for each edge $e_{ij} = (v_i, v_j) \in E$. In Eq. 6, the node features of v_i and v_j, i.e., $\boldsymbol{h}_i^{(l-1)}$, $\boldsymbol{h}_j^{(l-1)}$, along with the fattened coordinate difference ($\boldsymbol{z}_{ij}^{(l-1)}$), are concatenated and transformed by an MLP, generating the output edge feature for the l-th layer. Next, the coordinate module ϕ_X updates the node coordinates $\boldsymbol{X}_i^{(l)}$ using the updated edge feature $\boldsymbol{h}_{e_{ij}}^{(l)}$ and the coordinate differences $\boldsymbol{X}_{ij}^{(l-1)}$ in Eq. 7. Specifically, for each node v_i, ϕ_X first computes the message $\boldsymbol{m}_{j\to i}$ propagated from its neighbor v_j, and then updates the coordinates $\boldsymbol{X}_i^{(l)}$ of v_i by aggregating the messages from its neighborhood:

$$\boldsymbol{m}_{j\to i} = \text{MLP}\left(\boldsymbol{h}_{e_{ij}}^{(l)}\right) \cdot \boldsymbol{X}_{ij}^{(l-1)}, \tag{9}$$

$$\boldsymbol{X}_i^{(l)} = \boldsymbol{X}_i^{(l-1)} + \frac{1}{|\mathcal{N}_i|} \sum_{v_j \in \mathcal{N}_i} \boldsymbol{m}_{j\to i}. \tag{10}$$

Algorithm 1 Multi-channel Equivariant Graph Neural Network (MEGNN)

Input: Antibody CDR Structure $G = (V, E)$, initial features $\boldsymbol{h}_i^{(0)}$ and coordinates $\boldsymbol{X}_i^{(0)}$ for each node $v_i \in V$
Output: Antibody CDR representation $\boldsymbol{h}_G$
Initialize coordinates $\hat{\boldsymbol{X}}_i \leftarrow \boldsymbol{X}_i + \mathcal{N}(0, \sigma)$
for layer $l = 1$ **to** L **do**
  for $v_i \in V$ **do**
    for $v_j \in \mathcal{N}_i$ **do**
      Calculate the coordinate differences: $\boldsymbol{X}_{ij}^{(l-1)} \leftarrow \boldsymbol{X}_i^{(l-1)} - \boldsymbol{X}_j^{(l-1)}$
      Calculate the square distance: $\boldsymbol{z}_{ij}^{(l-1)} \leftarrow (\boldsymbol{X}_{ij}^{(l-1)})^\top \boldsymbol{X}_{ij}^{(l-1)}$
      Update the edge feature: $\boldsymbol{h}_{e_{ij}}^{(l)} \leftarrow \phi_e \left(\text{CONCAT} \left(\boldsymbol{h}_i^{(l-1)}, \boldsymbol{h}_j^{(l-1)}, \boldsymbol{z}_{ij}^{(l-1)} \right) \right)$
      Derive the propagated information: $\boldsymbol{m}_{j \leftarrow i} \leftarrow \text{MLP} \left(\boldsymbol{h}_{e_{ij}}^{(l)} \right) \cdot \boldsymbol{X}_{ij}^{(l-1)}$
    end for
    Update the coordinate: $\boldsymbol{X}_i^{(l)} \leftarrow \boldsymbol{X}_i^{(l-1)} + \frac{1}{|\mathcal{N}_i|} \sum_{v_j \in \mathcal{N}_i} \boldsymbol{m}_{j \to i}$
    Derive the aggregated edge feature: $\boldsymbol{h}_{agg_i}^{(l)} \leftarrow \sum_{j \in \mathcal{N}_i} \boldsymbol{h}_{e_{ij}}^{(l)}$
    Update the node representation: $\boldsymbol{h}_i^{(l)} \leftarrow \boldsymbol{h}_i^{(l-1)} + \text{MLP} \left(\text{CONCAT}(\boldsymbol{h}_i^{(l-1)}, \boldsymbol{h}_{agg_i}^{(l)}) \right)$
  end for
end for
Generate the representation of the input CDR structure: $\boldsymbol{h}_G \leftarrow \text{READOUT}(\{\boldsymbol{h}_i^{(L)}\}_{i=1}^n)$
Return: CDR representation $\boldsymbol{h}_G$

Finally, the node module ϕ_h updates the node representation $\boldsymbol{h}_i^{(l)}$ by Eq. 8. For each node v_i, ϕ_h aggregates the features of the adjacent edges into $\boldsymbol{h}_{agg_i}^{(l)}$ and combines the node representation $\boldsymbol{h}_i^{(l-1)}$ from the $(l-1)$-th layer with the aggregated feature using a residual connection (He et al., 2016):

$$\begin{aligned} \boldsymbol{h}_{agg_i}^{(l)} &= \sum_{j \in \mathcal{N}_i} \boldsymbol{h}_{e_{ij}}^{(l)}, \\ \boldsymbol{h}_i^{(l)} &= \boldsymbol{h}_i^{(l-1)} + \text{MLP} \left(\text{CONCAT}(\boldsymbol{h}_i^{(l-1)}, \boldsymbol{h}_{agg_i}^{(l)}) \right). \end{aligned} \tag{11}$$

CDR Embedding Generation. After the equivariant message passing through an L-layer MEGNN, we employ a READOUT function to aggregate the final node features to generate the representation of CDR G that consists of n nodes (amino acids) as Eq. 12,

$$\boldsymbol{h}_G = \text{READOUT}(\{\boldsymbol{h}_i^{(L)}\}_{i=1}^n). \tag{12}$$

The READOUT function can be a permutation invariant function, e.g. summation and element-wise mean pooling functions. In our implementation, we set the READOUT function as element-wise mean pooling by default. Algorithm 1 summarizes the forward pass of MEGNN.

C.2 Learning Objective and Sequence Generation

We train the MEGNN encoder by a self-supervised distance prediction task that explicitly aligns pairs of similar CDR in a given database. The goal is to align the structural representation of similar CDR pairs. For a CDR database $\mathcal{B} = \{G_1, G_2, \cdots, G_n\}$, we construct a training dataset $\mathcal{T} = \{(G_i, G_j), \cdots\}$ containing pairs of fixed-length CDRs whose TM-Score, calculated by TM-align (Zhang & Skolnick, 2005), exceeds a specified threshold. Given a pair of CDRs (G_i, G_j), we first generate their representations using MEGNN, denoted as $\boldsymbol{h}_{G_i}$ and $\boldsymbol{h}_{G_j}$, respectively. Next, we predict the *Root Mean Square Deviation (RMSD)* of the two CDR structures by feeding the concatenation of $\boldsymbol{h}_{G_i}$ and $\boldsymbol{h}_{G_j}$ into an MLP decoder as Eq. 13:

$$\widehat{d}(G_i, G_j) = \text{MLP} \left(\text{CONCAT} \left(\boldsymbol{h}_{G_i}, \boldsymbol{h}_{G_j} \right) \right). \tag{13}$$

Algorithm 2 CDR Vector Database Construction

Input: Training set $\mathcal{T} = \{(G_{i1}, G_{i2})\}_{i=1}^{|\mathcal{T}|}$, training epoch T, CDR database $\mathcal{B} = \{(\boldsymbol{s}_j, G_j)\}_{j=1}^{|\mathcal{B}|}$
Output: CDR vector database $\mathcal{Z}$
for $t = 1$ **to** T **do**
  for $i = 1$ **to** $|\mathcal{T}|$ **do**
    Initialize feature matrices $\boldsymbol{H}_{i1}$ of G_{i1} and feature matrices $\boldsymbol{H}_{i2}$ of G_{i2}, respectively
    Generate graph representation for the i-th training CDR pair: $\boldsymbol{h}_{G_{i1}} \leftarrow \text{MEGNN}(G_{i1}, \boldsymbol{H}_{i1}, \boldsymbol{X}_{i1})$, $\boldsymbol{h}_{G_{i2}} \leftarrow \text{MEGNN}(G_{i2}, \boldsymbol{H}_{i2}, \boldsymbol{X}_{i2})$
    Predict the RMSD between $h_{G_{i1}}$ and $h_{G_{i2}}$: $\widehat{d}(G_{i1}, G_{i2}) \leftarrow \text{MLP}\left(\text{CONCAT}\left(\boldsymbol{h}_{G_{i1}}, \boldsymbol{h}_{G_{i2}}\right)\right)$
    Compute the loss function: $\mathcal{L} \leftarrow \frac{1}{|\mathcal{T}|} \sum_{(G_{i1}, G_{i2}) \in \mathcal{T}} \|\widehat{d}(G_{i1}, G_{i2}) - d(G_{i1}, G_{i2})\|^2$
  end for
  Update the model weights $\boldsymbol{W}$ to minimize $\mathcal{L}$ using $\frac{\partial \mathcal{L}}{\partial \boldsymbol{W}}$
end for
for $j = 1$ **to** $|\mathcal{B}|$ **do**
  Generate graph representation $G_j \leftarrow \text{MEGNN}(G_j, \boldsymbol{H}_j, \boldsymbol{X}_j)$ Add the triplet $(\boldsymbol{s}_j, G_j, \boldsymbol{h}_{G_j})$ into $\mathcal{Z}$
end for
Return: Vector database $\mathcal{Z}$

Loss Function. The learning objective is to minimize the Mean Square Error between the predicted distance $\widehat{d}(G_i, G_j)$ and the actual distance $d(G_i, G_j)$ in the training dataset $\mathcal{T}$:

$$\mathcal{L} = \frac{1}{|\mathcal{T}|} \sum_{(G_i, G_j) \in \mathcal{T}} \|\widehat{d}(G_i, G_j) - d(G_i, G_j)\|^2. \tag{14}$$

Here, the actual distance $d(G_i, G_j)$ is computed as the RMSD of the two CDRs for their backbone atoms. Since we do not have prior knowledge of the CDR cluster labels, our approach can be interpreted as an unsupervised geometric learning model. By minimizing the loss function defined in Eq. 14, the model effectively generates CDR embeddings that reflect the structural relationships among the CDRs in the dataset.

CDR Sequence Generation. Once the model training is complete, we establish a CDR vector database $\mathcal{Z}$, where each CDR_i is represented by a triplet $(\boldsymbol{s}_i, G_i, \boldsymbol{h}_{G_i})$ consisting of amino acid sequence $\boldsymbol{s}_i$, its backbone structure graph G_i and its embedding $\boldsymbol{h}_{G_i}$ generated by the MEGNN encoder via Eq. 12. IgSeek is then able to infer the amino acid sequence of a CDR by querying its backbone structure in the database $\mathcal{Z}$. Let $\boldsymbol{s}_q$ denote the query CDR sequence with a length of L. At each position $l \in \{1, \cdots, L\}$, the residue $\boldsymbol{s}_q(l)$ is selected from one of the 20 amino acids, denoted as a_i for $i \in \{1, \cdots, 20\}$. Then, the inference of the CDR sequence $\boldsymbol{s}_q$ given its backbone structure G_q follows four steps: *(i)* first, the MEGNN encoder generates the embedding of G_q, denoted as $\boldsymbol{h}_{G_q}$. *(ii)* Second, the embedding $\boldsymbol{h}_{G_q}$ is used as the search key to perform a K-NN search in the database $\mathcal{Z}$, obtaining a set of K CDRs of equal length L, denoted as $\mathcal{Z}_q = \{(\boldsymbol{s}_1, G_1, \boldsymbol{h}_{G_1}), (\boldsymbol{s}_2, G_2, \boldsymbol{h}_{G_2}), \cdots, (\boldsymbol{s}_K, G_K, \boldsymbol{h}_{G_K})\}$. *(iii)* Given the K sequences $\mathcal{S}_q = \{\boldsymbol{s}_1, \cdots, \boldsymbol{s}_K\}$, we derive the probability of amino acid a_i occurring at position l of the predicted sequence $\hat{\boldsymbol{s}}_q$ as follows:

$$p\left(\hat{\boldsymbol{s}}_q(l) = a_i | S_q\right) = \frac{1}{K} \sum_{s_k \in \mathcal{S}_q} \mathbb{I}(\boldsymbol{s}_k(l), a_i),$$

where $\mathbb{I}(\boldsymbol{s}_k(l), a_i) \in \{0, 1\}$ is a binary indicator that equals 1 if the amino acid a_i occurs at the position l of sequence $\boldsymbol{s}_k$, and 0 otherwise. *(iv)* To derive the final inferred sequence $\hat{s}_q$, we sample the amino acid at each position l according to the generated probability distribution:

$$\hat{\boldsymbol{s}}_q(l) \sim p\left(\hat{\boldsymbol{s}}_q(l) | \mathcal{S}_q\right).$$

Algorithm 2 outlines the training process, and Algorithm 3 presents the antibody CDR sequence design process, respectively.

Algorithm 3 Sequence Generation

Input: Query structure G_q, MEGNN ϕ, CDR vector database $\mathcal{Z}$
Output: Predicted sequence $\hat{\boldsymbol{s}}_q$
Initialize feature matrix $\boldsymbol{H}_q$ and coordinates $\boldsymbol{X}_q$
Generate graph representation $G_q \leftarrow \text{MEGNN}(G_q, \boldsymbol{H}_q, \boldsymbol{X}_q)$
Retrieve the K-nearest neighbors of $\boldsymbol{h}_G$ in the database $\mathcal{Z}$ as $\mathcal{Z}_q$
Derive the probability of amino acid a at the l-th position: $p\left(\hat{\boldsymbol{s}}_q(l) = a_i | S_q\right) = \frac{1}{K}\sum_{s_k \in \mathcal{S}_q} \mathbb{I}(\boldsymbol{s}_k(l), a_i)$
Sample the amino acid $\hat{\boldsymbol{s}}_q(l)$ at the l-th position using the probability $p\left(\hat{\boldsymbol{s}}_q(l) | \mathcal{S}_q\right)$
Return: Sequence $\hat{\boldsymbol{s}}_q$

C.3 Analysis

Model Complexity. Given a 3D CDR structure represented by $G = (V, E)$, the initialized coordinates, node features, and the graph structure contribute a space complexity of $O(|V| \cdot d + |V| \cdot c + |E|) = O(|V| \cdot d + |E|)$, where d denotes the hidden dimension of features and c denotes the channel size. In MEGNN, the space complexity is dominated by the edge features, which have a complexity of $O(|E| \cdot d)$, and square distance $\boldsymbol{z}$ with a complexity of $O(|E| \cdot c^2)$. Consequently, the overall space complexity is $O(|E| \cdot d)$, which is linear to the input graph size. Regarding the computational complexity of MEGNN, the dominant component is the edge module ϕ_e introduced in Eq. 6, which has a time complexity of $O(|E| \cdot (2d + 3c)^2 + |E| \cdot d^2 + 3c) = O(|E| \cdot d^2)$.

Coordinate Equivariance and Representation Invariance. The following theorem shows that MEGNN is E(3) equivariant with respect to the initial coordinate $\boldsymbol{X}_i^{(0)}$ and E(3) invariant with respect to the representations $\boldsymbol{h}$ of the input CDR, respectively.

Theorem 1. *For any transformation $g \in E(3)$, we have $\boldsymbol{h}_i, T_{\mathcal{Y}}(g)\boldsymbol{X}_i^{(L)} = \text{MEGNN}\left(\boldsymbol{h}_i^{(0)}, T_{\mathcal{X}}(g)\boldsymbol{X}_i^{(0)}, G\right)$, where $T_{\mathcal{X}}$ and $T_{\mathcal{Y}} := \boldsymbol{R}\boldsymbol{X} + \boldsymbol{b}$ denotes the transformation of $\boldsymbol{X}$ in the input space $\mathcal{X}$ (resp. output space $\mathcal{Y}$), $\boldsymbol{R}$ is an orthogonal matrix, and $\boldsymbol{b}$ is the bias.*

The theorem indicates that MEGNN can be generalized to arbitrary E(3) group operations (refer to Section B), which showcases the data efficiency of MEGNN.

Proof. We assume that $\boldsymbol{h}_i^{(0)}$ is invariant to E(3) transformation operations on the coordinate $\boldsymbol{X}_i^{(0)}$, since $\boldsymbol{h}_i^{(0)}$ is generated from uniform distribution and no absolute information of $\boldsymbol{X}_i^{(0)}$ is encoded into $\boldsymbol{h}_i^{(0)}$. Then, for the E(3) transformation $g := \boldsymbol{R}\boldsymbol{X} + \boldsymbol{b}$, where orthogonal matrix $\boldsymbol{R} \in O(3)$ and bias $\boldsymbol{b} \in \mathbb{R}^3$, we have:

$$\begin{aligned} \boldsymbol{R}\boldsymbol{X}_i^{(l-1)} + \boldsymbol{b} - (\boldsymbol{R}\boldsymbol{X}_j^{(l-1)} + \boldsymbol{b}) &= \boldsymbol{R}\boldsymbol{X}_{ij}^{(l-1)}, \\ (\boldsymbol{R}\boldsymbol{X}_{ij}^{(l-1)})^\top \boldsymbol{R}\boldsymbol{X}_{ij}^{(l-1)} &= z_{ij}^{(l-1)}. \end{aligned}$$

Therefore, the output $z_{ij}^{(l-1)}$ of Eq. 5 is E(3) invariant to transformation g.

As for Eq. 6, since $\boldsymbol{h}_i$, $\boldsymbol{h}_j$, and $z_{ij}^{(l-1)}$ are invariant to E(3) transformation operations, we can derive that $\boldsymbol{h}_{e_{ij}}^{(l)}$ is E(3) invariant.

Next, we will prove Eq. 7 is E(3) equivariant.

$$\begin{aligned} \boldsymbol{R}\boldsymbol{X}_i^{(l-1)} + \boldsymbol{b} + \frac{1}{|\mathcal{N}_i|}\sum_{v_j \in \mathcal{N}_i} \text{MLP}\left(\boldsymbol{h}_{e_{ij}}^{(l)}\right) \cdot \boldsymbol{R}\boldsymbol{X}_{ij}^{(l-1)} &= \boldsymbol{R}\left(\boldsymbol{X}_i^{(l-1)} + \frac{1}{|\mathcal{N}_i|}\sum_{v_j \in \mathcal{N}_i} \boldsymbol{m}_{j \to i}\right) + \boldsymbol{b} \\ &= \boldsymbol{R}\boldsymbol{X}_i^{(l)} + \boldsymbol{b}. \end{aligned}$$

Therefore, we have proven that any E(3) transformation operations on $\boldsymbol{X}_i^{(l-1)}$ leads to the same E(3) transformation operations on $\boldsymbol{X}_i^{(l)}$ using Eq. 7.

Table 3: Hyperparameters of IgSeek.

Hyperparameter	Value	Description
		Input
noise_ratio	0.15	Ratio of the input coordinates with added Gaussian noise.
noise_scale	1	The standard deviation σ in the Gaussian noise.
θ	10 Å	The Euclidean distance threshold when constructing the graph G.
		MEGNN
learning_rate	5×10^{-3}	Learning rate of MEGNN.
weight_decay	1×10^{-4}	Weight decay factor of the optimizer.
hidden_dim	256	Size of hidden feature dimension in MEGNN.
emb_dim	128	Size of output embedding dimension in MEGNN.
n_layer	4	Number of layers in MEGNN.
epoch	50	Number of the iterations during training
batch_size	8	Number of batch size in MEGNN.
drop_out	0.1	Number of dropout rate in MEGNN.
		Retrieval
k	10	Number of nearest neighbor retrieved in the CDR vector database.
n_sample	2	Number of generated samples for each query.

Finally, it is easy to verify that Eq. 8 is E(3) invariant as $\boldsymbol{h}_i^{(l-1)}$ and $\boldsymbol{h}_{e_{ij}}^{(l)}$ are E(3) invariant.

In conclusion, for an L-layer MEGNN model, any transformation $g \in E(3)$ on the input coordinate $\boldsymbol{X}^{(0)}$ will lead to the same E(3) transformation operations on the output coordinate $\boldsymbol{X}^{(L)}$ while the representations $\boldsymbol{h}^{(L)}$ still remain E(3) invariant:

$$\boldsymbol{h}_i, T_{\mathcal{Y}}(g)\boldsymbol{X}_i^{(L)} = \text{MEGNN}\left(\boldsymbol{h}_i^{(0)}, T_{\mathcal{X}}(g)\boldsymbol{X}_i^{(0)}, G\right).$$

This finishes the proof. □

D Datasets and Labels

Datasets. We selected all experimentally solved antibody structures released in the SAbDab antibody database (Dunbar et al., 2013; Schneider et al., 2021) before January 1, 2024, to sample our training set. Notice that we remove CDR sequences that are identical to those in the dataset to eliminate redundancy in the dataset. Following FoldSeek (Van Kempen et al., 2024), for each CDR in the SAbDab-before-2024 dataset, we randomly sample equal-length CDRs with TM-score large than 0.6 to generate training pairs. The final training set consisted of $45,043$ antibody CDR pairs. After finishing model training, all $24,479$ unique CDR structures in the SAbDab-before-2024 dataset are utilized to construct the CDR vector database. The test set of SAbDab-2024 include experimentally solved antibody released in SAbDab antibody database between January 3, 2024 and May 29, 2024. This process resulted in $4,449$ test CDR samples that are completely unseen during the model training process.

The sequence similarity distribution between the training set and test set is illustrated in Figure 7. As we can observe, the average sequence similarity for each CDR region in the training and test set is around 0.3 to 0.5, which shows that there is no potential data leakage issue in this data split strategy. In addition, we utilize a T-cell receptor dataset released in the structural T-cell receptor database (Leem et al., 2018) to construct a test set with $5,111$ receptors, referred to as STCRDab. To evaluate the model efficiency, we utilize $5,000$ predicted CDR-H3 loops from the Observed Antibody Space (OAS) (Olsen et al., 2022), denoted as OAS-H3. Redundant CDR loops are removed from the test set. Statistics of these datasets are listed in Table 4.

Labels. PyIgClassify cluster labels (North et al., 2011; Adolf-Bryfogle et al., 2015) are employed as ground-truth labels to assess the retrieval performance of antibody CDR regions. For each PDB structure containing an identified antibody heavy or light chain, PyIgClassify categorizes the conformations of CDRs using a three-tier strategy: chain and position, length, and the similarity of dihedral angles. For instance, the cluster ID L1-10-1 denotes a CDR-L1 with a length of 10 amino

Table 4: Profile of Datasets

SAbDab	#CDR-H1	#CDR-H2	#CDR-H3	#CDR-L1	#CDR-L2	#CDR-L3
# Data (before-2024)	4,464	4,466	4,463	3,693	3,696	3,897
# Query (2024)	809	823	513	580	607	578
STCRDab	**#CDR-A1**	**#CDR-A2**	**#CDR-A3**	**#CDR-B1**	**#CDR-B2**	**#CDR-B3**
# Data	680	680	680	741	741	741
# Query	138	140	120	158	154	138

acids, where the subcluster 1 is determined based on the similarity of dihedral angles using the affinity propagation clustering method (Frey & Dueck, 2007).

E Implementation Details

In this section, we introduce the implementation details of our IgSeek. The MEGNN model introduced in Section C consists of three key learnable functions:

- The edge module ϕ_e (refer to Eq. 6) consists of a two-layer MLP with two Leaky Rectified Linear Unit (LeakyReLU) activation functions (Xu et al., 2015). Besides, a dropout function (Srivastava et al., 2014) with 0.1 dropout rate is employed on the output of ϕ_e:

  $$\text{CONCAT(Features)} \rightarrow \text{Input} \rightarrow \{\text{LinearLayer()} \rightarrow \text{LeakyReLU()} \rightarrow \text{LinearLayer()} \rightarrow \text{LeakyReLU()}\} \rightarrow \text{Dropout} \rightarrow \text{Output}.$$

- The coordinate module ϕ_X (refer to Eq. 7) contains a two-layer MLP that shares weights with the MLP in the edge module ϕ_e.
- The node module ϕ_h (refer to Eq. 8) is a two-layer MLP with one LeakyReLU activation function:

  $$\text{CONCAT(Features)} \rightarrow \text{Input} \rightarrow \{\text{LinearLayer()} \rightarrow \text{LeakyReLU()} \rightarrow \text{LinearLayer()}\} \rightarrow \text{Output}.$$

In our experiments, we train the MEGNN model in IgSeek using PyTorch (Paszke et al., 2019) with an Adam optimizer (Kingma & Ba, 2015) on 4 NVIDIA Tesla A100 GPUs. Table 3 lists the hyperparameters of IgSeek.

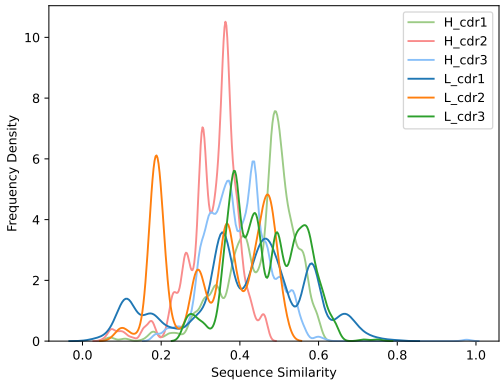

Figure 7: Sequence similarity between SAbDab train/test set.

F Baselines

The first category is structure retrieval model:

- **FoldSeek** (Van Kempen et al., 2024) represents tertiary amino acid interactions using 3D interaction (3Di) structural alphabet, achieving 4 to 5 orders of magnitude speed-up compared to traditional iterative or stochastic structure retrieval methods like CE (Shindyalov & Bourne, 1998), Dali (Holm, 2020), and TM-align (Zhang & Skolnick, 2005). Official code is available at: https://github.com/steineggerlab/foldseek.

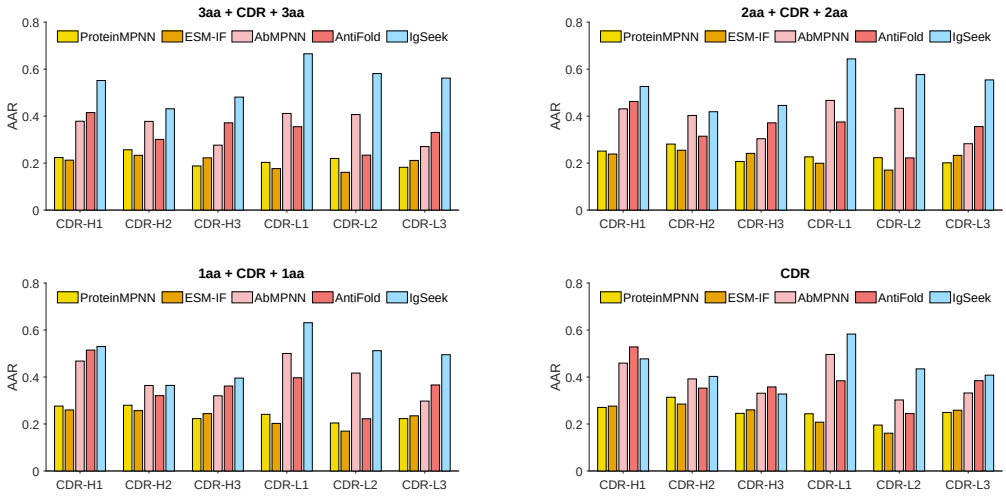

Figure 8: The Comparison of Average AAR on the SAbDab-2024 Dataset using CDRs with extensions of 1 to 3 amino acids on each side in the flanking regions.

The second category is protein and antibody design models:

- **ProteinMPNN** (Dauparas et al., 2022) is a deep learning–based method for protein sequence design that excels in both in silico and experimental evaluations, achieving a sequence recovery of 52.4% on native protein backbones, compared to 32.9% for Rosetta (Adolf-Bryfogle et al., 2018; Baek et al., 2021). By leveraging a message-passing neural network with enhanced input features and edge updates, ProteinMPNN is capable of designing monomers, cyclic oligomers, protein nanoparticles, and protein-protein interfaces, rescuing previously failed designs generated by Rosetta (Baek et al., 2021) or AlphaFold (Jumper et al., 2021). Official code is available at: https://github.com/dauparas/ProteinMPNN.
- **ESM-IF1** (Hsu et al., 2022) employs a sequence-to-sequence Transformer to predict protein sequences from backbone atom coordinates, which is pre-trained on structures of 12M protein sequences. It achieves 51% native sequence recovery and 72% for buried residues. Official code is available at: https://github.com/facebookresearch/esm/tree/main/examples/inverse_folding.
- **AbMPNN** (Dreyer et al., 2023) fine-tunes ProteinMPNN on the SAbDab (Dunbar et al., 2013; Schneider et al., 2021) dataset for antibody design, outperforming generic protein models in sequence recovery and structure robustness, especially for the hypervariable CDR-H3 loop. The profile of model weights is available at: https://zenodo.org/records/8164693.
- **AntiFold** (Høie et al., 2024) is an antibody-specific inverse folding model fine-tuned from ESM-IF1 (Hsu et al., 2022) on solved antibody structures from the SAbDab dataset (Dunbar et al., 2013; Schneider et al., 2021) and predicted antibody structures from the OAS dataset (Kovaltsuk et al., 2018; Olsen et al., 2022). AntiFold excels in sequence recovery and structural similarity while also demonstrates stronger correlations in predicting antibody-antigen binding affinity in a zero-shot manner. Official code is available at: https://github.com/oxpig/AntiFold.

G Additional Experiments

CDR with extensions. In this set of experiments, we compare IgSeek with protein and antibody design baselines using the SAbDab-2024 dataset. We focus on CDRs with backbone extensions of n amino acids on each side in the flanking regions. Fig. 8 illustrates the results for varying values of $n = 0, 1, 2, 3$. As we can observe, the performance of IgSeek improves with the inclusion of additional amino acids in the given structure, , which aligns with the fact that more input structural information can be encoded into the CDR representation. In contrast, other baseline models are adversely affected by hallucinations stemming from conserved backbone structures. Notably, when $n = 3$, IgSeek consistently outperforms its competitors by at least 5% and 18% for heavy chain and light chain CDR loops, respectively. This further demonstrates that the retrieval-based strategy employed by IgSeek effectively mitigates hallucinations during CDR sequence generation.

Table 5: The Comparison of Average AAR with varying K in SAbDab-2024.

K	5	10	20	50	100
CDR-L1	0.660	0.658	0.645	0.620	0.593
CDR-L2	0.580	0.580	0.573	0.573	0.550
CDR-L3	0.586	0.586	0.576	0.574	0.564
CDR-H1	0.560	0.561	0.560	0.553	0.537
CDR-H2	0.440	0.435	0.432	0.429	0.430
CDR-H3	0.473	0.464	0.455	0.447	0.441

Influence of value K. In this set of experiments, we conduct experiments on the SAbDab-2024 dataset to evaluate the impact of varying parameter K in IgSeek. Table 5 reports the average AAR of IgSeek across different values of K on the SAbDab-2024 dataset. As we can observe, the performance of IgSeek exhibits a decline as K increases. In our implementation, we set $K = 10$ rather than 5 as IgSeek achieves comparable results while preserving enhanced sequence diversity.