Improve Antibody Humanization with Monte-Carlo Tree Search

Zhihao Zhan [1 2] **Huan Zhang** [1 2]

Abstract

Antibody humanization is a critical step in developing therapeutic antibodies, aimed at reducing immunogenicity while preserving binding specificity and affinity. Traditional computational approaches, such as BioPhi, rely on language models to predict humanized antibody sequences by iteratively mutating non-human-like amino acids. However, the inference-time scalability of language-model-based antibody humanization prediction remains underexplored. In this work, we propose an optimized Monte Carlo Tree Search (MCTS) framework to enhance antibody humanization. We formulate the task as an MCTS problem, where states represent antibody sequences, actions correspond to single amino acid mutations, and rewards are derived from predictions made by pre-trained language models. To address the challenges posed by the large state and action spaces, we introduce key optimizations to enhance standard MCTS, including state and action deduplication, heuristic-driven expansion, and modifications to streamline the tree iteration process. Our experiments demonstrate that the optimized MCTS outperforms both BioPhi and standard MCTS, producing humanized antibody sequences with superior drug-likeness, human antibody-likeness, and human germline-likeness. This approach significantly enhances scalability and accuracy, paving the way for more efficient in silico antibody design.

1. Introduction

Recent years have witnessed the rise of therapeutic proteins. Compared to small molecule drugs, these proteins generally offer improved target specificity, reduced toxicity, and extended durations of action within human metabolism (Jarasch et al., 2015). Antibodies and their derivatives represent an important class of therapeutic proteins with widespread applications in areas such as cancer treatment and antigen detection. To date, more than 100 antibody-based drugs have been approved in the United States and Europe, and this number continues to grow at an increasing rate each year (The Antibody Society, 2023).

A standard antibody drug development pipeline includes four stages: hit generation, lead selection, lead optimization, and clinical trials. This entire process is highly demanding, typically costing more than 1 billion USD and taking more than 10 years to complete, with a low success rate of approximately 4% (Gaudelet et al., 2021). Since nearly all efforts in the clinical stage focus solely on one or a few selected antibodies, the thoroughness and accuracy of early-stage experiments play a decisive role in the success or failure of antibody drug development.

The advancement of computer science and artificial intelligence has driven continuous progress in the use of deep language models to predict protein properties. Proteins are represented as amino acid sequences that can be effectively modeled using language models. Large-scale models, with billions of parameters and trained on millions of protein sequences, are capable of transferring learned knowledge to a range of downstream prediction tasks (Rives et al., 2021; Lin et al., 2022; Hayes et al., 2024). Although the predictive accuracy of these models may not match the precision of biological experiments, their low cost and high efficiency make them ideal for large-scale preliminary screening of protein sequences, thereby enhancing the success rate of experimental validation in wet-lab settings.

Antibody humanization is a crucial process during lead optimization. Prior to this stage, the ability of the lead antibody to bind effectively to the target was validated. The goal of antibody humanization is to generate a molecule with reduced immunogenicity in humans, while maintaining the specificity and binding affinity of the original nonhuman antibody (Almagro & Fransson, 2008).

Conducting in silico antibody humanization is a challenging task due to two primary obstacles. The first challenge is the lack of data. There are few successfully developed antibodies, meaning we may not have a reliable supervision signal for direct model training. As a result, it requires humanization models capable of transferring correlated knowledge

[1]Mila - Québec AI Institute [2]Université de Montréal. Correspondence to: Zhihao Zhan <zhihao.zhan@mila.quebec>, Huan Zhang <huan.zhang@mila.quebec>.

IFT 6269 project report - Université de Montréal

to this task in a zero-shot or few-shot setting. The second challenge lies in the complexity of the humanization task itself. Recently, BioPhi (Prihoda et al., 2022) has addressed these issues. The researchers first pre-trained an antibody language model using masked language modeling on the Observed Antibody Space (OAS) (Kovaltsuk et al., 2018), which contains over five hundred million human sequences from more than five hundred individuals. They then framed antibody humanization as a mutation process: first, the language model takes an antibody sequence as input and predicts the probability of each amino acid type at every position. A humanized sequence is then generated by concatenating the amino acid type with the highest probability at each position. The underlying intuition of this process is that the language model is trained on the data distribution of human antibodies, so a high probability for a particular amino acid at a given position reflects its likelihood of being part of a human antibody sequence. In practice, the mutation process is conducted iteratively for 1 to 4 cycles.

In this work, we extend BioPhi with inference-time scaling. Empirically, we show that BioPhi's decoding algorithm is not scalable since it can get stuck at a local optimal solution in a few iterations. These days, inference-time scalable algorithms have shown significant potential in enhancing language modeling performance (Snell et al., 2024), particularly for complex reasoning tasks (OpenAI, 2024). Monte Carlo Tree Search (MCTS) has proven to be one of the most effective methods for inference-time planning (Yao et al., 2023; Hao et al., 2023; Feng et al., 2024b). In light of this, we reformulate antibody humanization as an MCTS problem. We define a state as an antibody sequence, an action as the mutation of single amino acid, and calculate rewards from the probabilities predicted by the pre-trained language model. However, we find that the trivial application of standard MCTS to antibody humanization cannot run efficiently and perform well owing to the large state and action space of the MCTS formulation. We further optimize MCTS from 3 different perspectives: state and action de-duplication, introducing heuristics for MCTS expansion, and eliminating the tree-climbing process of MCTS selection and back-propagation. In our experiments, we show that our optimized MCTS can scale better than BioPhi and standard MCTS and predict humanized sequences with improved drug likeliness, human antibody likeliness and human germline likeliness. Through ablation study, we show that both the introduced heuristics and efficiency improvements have positive impacts on the experiment results.

2. Related Work

2.1. Monte-Carlo Tree Search (MCTS)

Monte Carlo Tree Search (MCTS) is a versatile algorithm for decision-making under uncertainty, balancing exploration and exploitation in its search for optimal solutions. By combining simulation-based evaluation with a tree-based search framework, MCTS dynamically analyzes large search spaces, making it particularly effective for problems like game-playing and complex optimization tasks. The algorithm operates iteratively, building a search tree guided by random simulations (playouts). Each iteration comprises four steps:

1. **Selection:** Traversing the search tree from the root node using a selection policy, such as UCB1, to balance exploration and exploitation.
2. **Expansion:** Adding one or more child nodes to the tree based on untried actions.
3. **Simulation:** Conducting a random playout (or rollout) from a newly expanded node to a terminal state.
4. **Backpropagation:** Updating the nodes along the path to the root based on the result of the simulation.

MCTS has found extensive applications across various domains. In game-playing, it serves as a foundational component of systems like AlphaGo and AlphaZero, where it integrates with neural networks to achieve superhuman performance in games such as Go, Chess, and Shogi. By using neural networks to guide node selection and evaluation, these systems enhance the scalability and precision of the search process (Feng et al., 2024a). Beyond game-playing, MCTS has been employed in optimization problems such as scheduling, resource allocation, and robotics motion planning, where its ability to balance computational resources and focus on promising regions of the search space has proven invaluable.

Despite its versatility, standard MCTS faces challenges in scalability when dealing with high-dimensional tasks due to the computational cost of traversing large state-action spaces. To address these limitations, numerous variants and optimizations have been proposed. Heuristic-driven expansion leverages domain-specific insights to prioritize node exploration, as demonstrated in AlphaZero, where a policy network focuses the search on moves with higher probabilities of success (Feng et al., 2024a). State-action deduplication reduces redundancy in the search tree by identifying equivalent states and actions, thereby enhancing computational efficiency (Hao et al., 2023). Other modifications include parallelization, optimized rollout strategies, and the integration of machine learning techniques. For example, Neural-MCTS combines MCTS with neural networks for richer evaluations and action priors, leading to improved performance in decision-making tasks (Yao et al., 2023). In applications like antibody humanization, further optimizations involve simplifying the tree traversal process, such as

eliminating tree climbing during selection and backpropagation, to enhance efficiency without sacrificing decision quality (Feng et al., 2024a; Yao et al., 2023).

2.2. Protein Language Models

Protein language models are large-scale models pre-trained on protein sequence data. The model's high-quality sequential representations encapsulate key information about proteins, including their physical and chemical properties, homology, secondary and tertiary structures, and even certain mutation characteristics, has demonstrated strong transfer capability across different tasks (Rives et al., 2021; Lin et al., 2022; Hayes et al., 2024). For instance, Thumuluri et al. 2022 leveraged it to predict protein solubility and expression levels; Kilinc et al. 2023 employed it to identify protein homology and novel functions; Høie et al. 2022 utilized it for rapid protein structure prediction; and Brandes et al. 2022 applied it to forecast pathogenic protein mutations. Notably, in the context of antibody property prediction, Feng et al. 2022 used the protein language model ESM1b to predict antibody solubility, achieving promising results.

3. Method

3.1. Formulation

Typically, an antibody molecule consists of 4 peptide chains, of which 2 are heavy chains and the other 2 are light chains. The antibody molecule has a symmetric Y-shaped structure: The 2 heavy chains are identical, and the 2 light chains are also identical. Each peptide chain can be further divided into a variable region and a constant region. The constant regions of the different antibody peptide chains in the human body are the same, while the variable regions differ. Within the variable region, complementarity-determining regions (CDRs) are the sites where the antibody binds specifically to the antigen. Prior to the antibody humanization stage, the CDR regions have been verified to be effective, so the goal of antibody humanization is to optimize the non-CDR regions by replacing the non-human-like sub-regions to their human-like counterparts, in order to reduce immunogenicity.

In this work, we formulate this process as an iterative single amino acid mutation process. Given a reward function R that is based on a language model trained on human antibodies, starting with a sequence $s \in S^n$, where S is the set of 20 amino acid types and n is the length of the sequence, our goal is to design an algorithm to find a sequence of single amino acid mutation steps $(a_1, \ldots, a_l)$ that maximizes $R(a_l \circ \ldots \circ a_1(s))$, where $a_k \in \{f_{p,X} | p \in P_s, X \in S\}$, P_s is the non-CDR amino acids in the sequence based on the IMGT numbering of s (Giudicelli & Lefranc, 1999),

Algorithm 1 BioPhi

Input: antibody sequence s, language model LM
Initialize $\hat{s} \leftarrow s$
for $i = 1$ **to** $steps$ **do**
  $R(s), \mathbf{r}(s) \leftarrow \text{LM}(s)$
  if $R(s) > R(\hat{s})$ $\hat{s} \leftarrow s$
  $a_1, \ldots, a_{|P_s|} \leftarrow \{f_{i,\text{argmax}_X r_{i,X}} | i \in P_s\}$
  $s \leftarrow a_1 \circ \ldots \circ a_{|P_s|}(s)$
end for
return $\hat{s}$

Algorithm 2 MCTS

Input: antibody sequence s, language model LM
Initialize Tree $\leftarrow \varnothing$, Root $\leftarrow s$
Initialize $\hat{s} \leftarrow s$
$R(s), \mathbf{r}(s) \leftarrow \text{LM}(s)$
Tree.Expand($s, R(s)$,parent=NULL)
for $i = 1$ **to** $steps$ **do**
  $s_i \leftarrow$ Tree.Select(Root)
  [Note: Starting from Root, iteratively select a child until reaching a leaf s_i.]
  Sample a from $f_{p,X}(s_i)$
  $s' \leftarrow a(s_i)$
  $R(s'), \mathbf{r}(s') \leftarrow \text{LM}(s')$
  if $R(s') > R(\hat{s})$ $\hat{s} \leftarrow s'$
  Tree.Expand($s', R(s')$,parent=s_i)
  Tree.Update($s', R(s')$)
  [Note: Starting from s', iteratively update the expected future reward until reaching the Root.]
end for
return $\hat{s}$

$P_s \subset \{1, 2, \ldots, n\}$,

$$f_{p,X}(s) = s', s'_i = \begin{cases} s_i & i \neq p \\ X & i = p \end{cases}$$

represents a single amino acid mutation. We choose

$$R(s) = \sum_i r_{i,s_i}(s)$$

where $r_{p,X}(s)$ is the probability of the p-th amino acid being X predicted by the language model. The reason here is that if the language model is more certain that the p-th amino acid should be X, it indicates that a sequence with the p-th amino acid being X is more like a human antibody sequence, so that it would be more likely to be adapted by human body and less likely to trigger human immune response, so that the antibody drug can be more effective. Intuitively, there is a negative correlation between $R(s)$ and immunogenicity.

Using these annotations, the greedy BioPhi (Prihoda et al., 2022) algorithm can be reformulated as Algorithm 1, and the

Algorithm 3 Optimized MCTS

Input: antibody sequence s, language model LM
Initialize Tree $\leftarrow \varnothing$, Root $\leftarrow s$
Initialize $\hat{s} \leftarrow s$
$R(s), \mathbf{r}(s) \leftarrow \text{LM}(s)$
$\{s'\} \leftarrow \{a(s) | a \in f_{p,X}\}$
$L_s \leftarrow$ Sort $\{s'\}$ by $\{\tilde{R}_{s'}\}$
[Note: $\tilde{R}(L_{s,1}) > \tilde{R}(L_{s,2}) > \ldots$]
Heap.Push($L_{s,1}$)
[Note: max heap using $\tilde{R}(L_{s,1})$ as key.]
for $i = 1$ **to** $steps$ **do**
  $L_{p_i} \leftarrow$ Heap.Pop()
  [Note: p_i is the parent for the i-th step.]
  $s_i \leftarrow L_{p_i}$
  [Note: s_i is the child for the i-th step.]
  $R(s_i), \mathbf{r}(s_i) \leftarrow \text{LM}(s_i)$
  if $R(s') > R(\hat{s})$ $\hat{s} \leftarrow s'$
  $\{s'_i\} \leftarrow \{a(s_i) | a \in f_{p,X}\}$
  $L_{s_i} \leftarrow$ Sort $\{s'_i\}$ by $\{\tilde{R}_{s'_i}\}$
  [Note: $\tilde{R}(L_{s_i,1}) > \tilde{R}(L_{s_i,2}) > \ldots$]
  while $L_{s_i} \neq \varnothing$ and $L_{s_i,1}$ has been searched **do**
    L_{s_i}.PopFront()
  end while
  if $L_{s_i} \neq \varnothing$ Heap.Push($L_{s_i,1}$)
  L_{p_i}.PopFront()
  [Note: Current $L_{p_i,1}$ has been searched.]
  while $L_{p_i} \neq \varnothing$ and $L_{p_i,1}$ has been searched **do**
    L_{p_i}.PopFront()
  end while
  if $L_{p_i} \neq \varnothing$ Heap.Push($L_{p_i,1}$)
  [Note: Update new $L_{p_i,1}$ in heap.]
end for
return $\hat{s}$

standard MCTS algorithm can be reformulated as Algorithm 2.

3.2. Optimized MCTS

We argue that standard MCTS has the following limitations:

- **Duplicated states and actions.** Here, we provide two examples. The first example considers mutating "ASGF" to "VSGY". There are two possible MCTS search paths: "ASGF" $\rightarrow$ "VSGF" $\rightarrow$ "VSGY" and "ASGF" $\rightarrow$ "ASGY" $\rightarrow$ "VSGY". Standard MCTS treats the two "VSGY" states reached from these paths as different, leading to duplicated searches. Note that the duplicated computation grows exponentially as the number of mutations increases. Another example involves the mutation paths "ASGF" $\rightarrow$ "ASGY" $\rightarrow$ "ASGC". Here, mutating the last letter to "C" is redundant for the state "ASGY" since it can be directly obtained from "ASGF" $\rightarrow$ "ASGC".
- **Random expansion.** In standard antibody humanization setups, the action space is approximately $19n$, where $n \sim 100$ for common antibodies, and $19 = 20 - 1$ represents the number of amino acid types. This large discrete action space requires substantial computational resources to explore thoroughly through random MCTS expansion.
- **Inefficient tree iteration.** The selection and back-propagation stages in standard MCTS require visiting the path from the root to a leaf in the tree. This process becomes increasingly inefficient as the tree depth increases.

Therefore, we ask:

- *Can we avoid searching the same state more than once?*
- *Can we improve the expansion process with heuristics?*
- *Can we directly access the best node for expansion?*

In this section, we provide solutions to all these problems. To begin with, we slightly modify the objective of MCTS. We propose maintaining the maximal future reward instead of the expected future reward during expansion. This adjustment is reasonable as we are seeking the optimal humanized sequence.

Based on this modification, we improve standard MCTS in the following ways:

Deduplication. We enforce the following rules:

- If a newly expanded state has been encountered before, it is discarded.
- Once an action is performed at a position, future actions can no longer modify that position.

Heuristic Expansion. We calculate the estimated reward of the next state instead of the current state. Specifically, for $s' = a(s), a \in f_{p,X}$, we calculate:

$$\tilde{R}_{s'} = \sum_i r_{i,s'_i(s)}$$

. During expansion, we enumerate all possible s' and prioritize expanding those with high $\tilde{R}_{s'}$.

Eliminating selection and back-propagation. We use a heap to maintain the state with the maximal unvisited next state estimated reward. For selection, we directly access the heap's top. Back-propagation is no longer necessary; we

simply update the estimated reward of the current state in the heap.

Our method is formulated in Algorithm 3.

4. Experiments

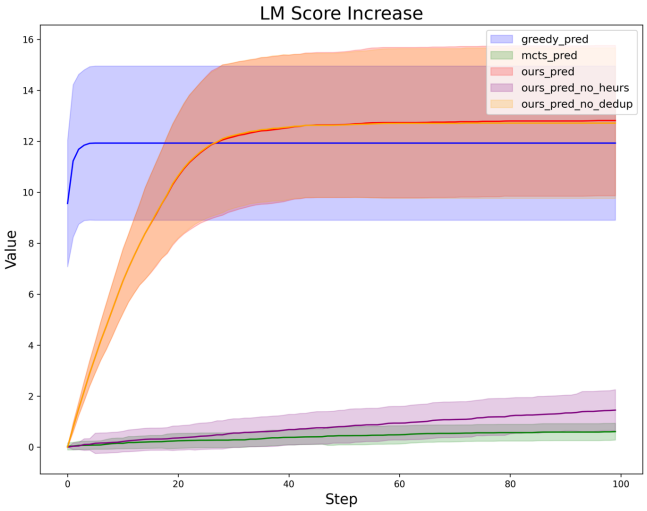

Figure 1. Performance comparison of our method against the greedy baseline, standard MCTS baseline, our method without heuristics, and our method without deduplication on the Hu-mAb dataset.

4.1. Setup

We use the Hu-mAb dataset for evaluation, which contains 25 pre-humanized antibodies. We compare our method with the greedy BioPhi (Prihoda et al., 2022) algorithm and standard MCTS. We also consider two ablation versions of our method: one without deduplication and one without heuristic search. We visualize the curves of LM score increase with the growth of search steps. Additionally, we report drug-likeness (OASis percentile), human antibody-likeness (OASis Identity), and human germline-likeness (Germline content percentile) evaluated under the default setting using the BioPhi platform. The results are summarized in Table 1 and Figure 1.

4.2. Results

We find that our method outperforms all baselines and ablation variants on all metrics, demonstrating its effectiveness. From the curves of LM score increase, we observe that the greedy BioPhi algorithm quickly reaches a plateau within just a few steps, as it gets stuck in a local maximum. Standard MCTS cannot efficiently achieve a high score due to the inefficiency of random expansion. When enhanced with our heuristic strategies, we observe a significant performance boost.

5. Discussion

Despite the promising results, several limitations warrant consideration. First, the current method focuses solely on mutation operations, excluding insertions and deletions, which limits the exploration of a broader range of sequence modifications. Incorporating these operations could enable more flexible sequence optimization. Second, the reward model's accuracy may be limited, as it relies on zero-shot predictions from pre-trained language models rather than a model specifically trained on experimental immunogenicity data. Building a regression model fine-tuned on such data could significantly improve evaluation reliability. However, the current lack of sufficient publicly available experimental data poses a challenge to this approach. Lastly, the heuristics employed in MCTS may cause the algorithm to converge on local maxima, reducing its ability to explore alternative solutions. In future work, we aim to investigate alternative approaches for incorporating effective randomness to enhance the algorithm's adaptability and overall performance.

References

Almagro, J. C. and Fransson, J. Humanization of antibodies. *Frontiers in Bioscience*, 2008.

Brandes, N., Goldman, G., Wang, C. H., Ye, C. J., and Ntranos, V. Genome-wide prediction of disease variants with a deep protein language model. *bioRxiv Preprint*, 2022.

Feng, J., Jiang, M., Shih, J., and Chai, Q. Antibody apparent solubility prediction from sequence by transfer learning. *iScience*, 2022.

Feng, X., Wan, Z., Wen, M., McAleer, S. M., Wen, Y., Zhang, W., and Wang, J. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*, 2024a. URL https://arxiv.org/abs/2309.17179.

Feng, X., Wan, Z., Wen, M., McAleer, S. M., Wen, Y., Zhang, W., and Wang, J. Alphazero-like tree-search can guide large language model decoding and training. *Proceedings of the International Conference on Machine Learning*, 2024b.

Gaudelet, T., Day, B., Jamasb, A. R., Soman, J., Regep, C., Liu, G., Hayter, J. B. R., Vickers, R., Roberts, C., Tang, J., Roblin, D., Blundell, T. L., Bronstein, M. M., and Taylor-King, J. P. Utilizing graph machine learning within drug

Table 1. a performance comparison on the Hu-mAb dataset, including ablation study.

METHOD	DRUG LIKELINESS	HUMAN ANTIBODY LIKELINESS	HUMAN GERMLINE LIKELINESS	LM SCORE INCREASE
BIOPHI	59.66	83.44	84.67	11.93
MCTS	7.33	52.63	69.50	0.61
OURS	**62.39**	**84.68**	**87.04**	**12.81**
- DEDUPLICATION	62.26	84.62	87.02	12.72
- HEURISTICS	4.72	49.24	69.22	1.45

discovery and development. *Briefings in Bioinformatics*, 2021.

Giudicelli, V. and Lefranc, M.-P. Ontology for immunogenetics: the imgt-ontology. *Bioinformatics*, 1999.

Hao, S., Gu, Y., Ma, H., Hong, J. J., Wang, Z., Wang, D., and Hu, Z. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. URL https://arxiv.org/abs/2305.14992.

Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M., et al. Simulating 500 million years of evolution with a language model. *bioRxiv*, 2024.

Høie, M. H., Kiehl, E. N., Petersen, B., Nielsen, M., Winther, O., Nielsen, H., Hallgren, J., and Marcatili, P. Netsurfp-3.0: accurate and fast prediction of protein structural features by protein language models and deep learning. *Nucleic Acids Research*, 2022.

Jarasch, A., Koll, H., Regula, J. T., Bader, M., Papadimitriou, A., and Kettenberger, H. Developability assessment during the selection of novel therapeutic antibodies. *Journal of Pharmaceutical Sciences*, 2015.

Kilinc, M., Jia, K. K., and Jernigan, R. L. Improved global protein homolog detection with major gains in function identification. *Proceedings of the National Academy of Sciences of the United States of America*, 2023.

Kovaltsuk, A., Leem, J., Kelm, S., Snowden, J., Deane, C. M., and Krawczyk, K. Observed antibody space: a resource for data mining next-generation sequencing of antibody repertoires. *The Journal of Immunology*, 2018.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv*, 2022.

Prihoda, D., Maamary, J., Waight, A., Juan, V., Fayadat-Dilman, L., Svozil, D., and Bitton, D. A. Biophi: A platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning. *mAbs*, 2022.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 2021.

Snell, C., Lee, J., Xu, K., and Kumar, A. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

OpenAI. Learning to reason with llms. Technical report, 2024. https://openai.com/index/learning-to-reason-with-llms/.

The Antibody Society. Therapeutic monoclonal antibodies approved or in review in the eu or us. Technical report, 2023. https://www.antibodysociety.org/resources/approved-antibodies.

Thumuluri, V., Martiny, H. M., Almagro Armenteros, J. J., Salomon, J., Nielsen, H., and Johansen, A. R. Netsolp: predicting protein solubility in escherichia coli using language models. *Bioinformatics*, 2022.

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. 2023. URL https://arxiv.org/abs/2305.10601.