# Harnessing Preference Optimisation in Protein LMs for Hit Maturation in Cell Therapy

**Katarzyna Janocha** **Annabel Ling** **Alice Godson** **Yulia Lampi**
**Simon Bornschein** **Nils Y. Hammerla**
Coding Bio
{kasia,annabel,alice,yulia,simon,nils}@coding.bio

## Abstract

Cell and immunotherapy offer transformative potential for treating diseases like cancer and autoimmune disorders by modulating the immune system. The development of these therapies is resource-intensive, with the majority of drug candidates failing to progress beyond laboratory testing. While recent advances in machine learning have revolutionised areas such as protein engineering, applications in immunotherapy remain limited due to the scarcity of large-scale, standardised datasets and the complexity of cellular systems. In this work, we address these challenges by leveraging a high-throughput experimental platform to generate data suitable for fine-tuning protein language models. We demonstrate how models fine-tuned using a preference task show surprising correlations to biological assays, and how they can be leveraged for few-shot hit maturation in CARs. This proof-of-concept presents a novel pathway for applying ML to immunotherapy and could generalise to other therapeutic modalities.

## 1 Introduction

Cell and immunotherapy hold tremendous promise for treating traditionally incurable diseases, with applications ranging from oncology and autoimmune disorders to age-related conditions. The overarching goal is to modulate the patient's immune response, either by directly engineering immune cells or by introducing engineered proteins to enhance or suppress immune function. The main modality we will explore in this work are Chimeric Antigen Receptors (CARs) - engineered proteins that are expressed on the surface of T-cells. CARs act as artificial receptors designed to bind to a specific antigen (target), for instance on the cell surface of a tumor, and trigger an immune response.

The development of immunotherapies, and drug development in general, is a resource-intensive process. Despite extensive efforts to evaluate candidates in the lab, the great majority of identified drug compounds will fail to progress through clinical trials to become viable treatments for patients [Mullard, 2016]. To improve the performance of candidate drug compounds, the field of drug discovery has increasingly turned to computational approaches to make better use of experimental data and facilitate exploration of the vast design space for such molecules.

Over recent years, the modelling of discrete time-series such as natural text has seen tremendous progress, where the capabilities of transformer-based language models [Vaswani, 2017] far exceeded the expectations by domain experts. Naturally, these methods have been applied to protein engineering, aiming to construct biologically relevant representations of amino-acid (or tokenised DNA-) sequences via masking [Lin et al., 2023, Hayes et al., 2024] or auto-regressive Protein Language Models (PLMs) [Nijkamp et al., 2023]. Areas such as structure prediction Abramson et al. [2024], structure generation [Watson et al., 2023], and protein-protein interactions such as docking [Corso et al., 2022] have seen significant progress via large and sophisticated ML models based on transformers or diffusion generative models, among other approaches. This progress is largely spurred by the availability of

AI for New Drug Modalities at NeurIPS 2024.

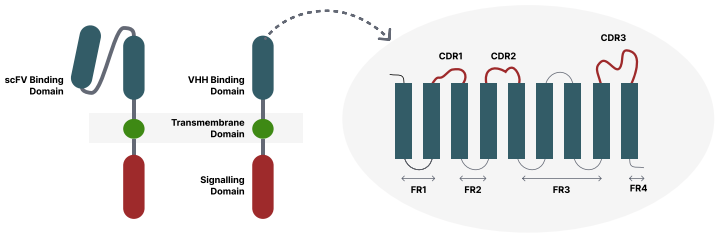


Figure 1: CAR structure with scFV and VHH binding domains (see text for details).

relevant data at scale [Suzek et al., 2015, Steinegger and Söding, 2018] and clearly defined research questions, which has led to a rich and mature ecosystem of academic and commercial interest around ML applications in protein engineering.

Another area of focus has been around in-silico design and optimisation of antibodies [Joubbi et al., 2024], with large datasets available for academic and commercial use [Olsen et al., 2022]. Applications include the virtual screening of potential binders [Bachas et al., 2022], affinity maturation [Ruffolo et al., 2021, Clark et al., 2023, Hie et al., 2024] or the optimisation of enzymes [He et al., 2024]. Much of the work in this space is driven by the availability of language models trained on antibody data, including masking [Ruffolo et al., 2021, Tobias H. Olsen and Deane, 2022, Prihoda et al., 2022] as well as auto-regressive models [Shuai et al., 2021, Nijkamp et al., 2023].

In contrast, ML applications in immunotherapy are relatively scarce, and have predominantly relied on basic ML such as motif-based optimisation of CARs [Castellanos-Rueda et al., 2022, Daniels et al., 2022], with more sophisticated modelling only being explored recently for bispecific antibodies [Mullin et al., 2024]. This is partly due to the lack of publicly accessible, standardized datasets, as well as the inherent complexity of the problem compared to small molecule drug design. The data in this field are influenced by a multitude of factors related to living cells, making it challenging to generate datasets at a scale sufficient for effective training or fine-tuning of large models. The field thus continues to rely on manual, predominantly low-throughput approaches to drug discovery.

In this work we attempt to bridge this gap, demonstrating how large-scale, low-precision yet high throughput testing of thousands of drug compounds in cells can be harnessed to generate data suitable for fine-tuning PLMs for the task of hit maturation in CARs. We show, perhaps surprisingly, that model loss of an auto-regressive PLM, fine-tuned using a preference task, is highly correlated to the results of standard biological assays, and that we can reliably find improved mutants via few-shot exploration of the design space. This provides a proof-of-concept that such models can effectively guide the design of CARs, which could generalise to other modalities in immunotherapy.

## 2 Chimeric Antigen Receptors (CARs)

Briefly, immunotherapies use a variety of mechanisms to harness the immune system to recognise and eliminate diseases. These immunotherapies include monoclonal antibodies (mAb), checkpoint inhibitors, T cell engagers and CAR-T cell therapy. CAR-T cell therapy has predominantly been used to treat haematological malignancies such as multiple myeloma and leukemia, where it may lead to complete remission in a significant fraction of patients [Cappell and Kochenderfer, 2023]. It also shows potential in other malignant diseases and autoimmune diseases [Blache et al., 2023].

Typically, a CAR consists of a binding domain which binds to antigens on tumour cells, a hinge domain that gives the binder flexibility, a transmembrane domain that anchors the CAR to the T cell membrane, and a signalling domain which facilitates T cell activation (Figure 1).

Various types of binding domains can be presented on the CAR, but most common are those based on single-chain fragment variable (scFv) of mAb. In contrast, here we explore the use and optimisation of variable heavy domain of heavy chain (VHH) fragments, which are antibody fragments naturally derived from the Camelidae family. scFvs and VHH fragments are comparable in terms of affinity [Muyldermans, 2013] yet the reduced size of VHHs promotes advantageous stability, solubility and immunogenicity characteristics [Gorovits and Koren, 2019].

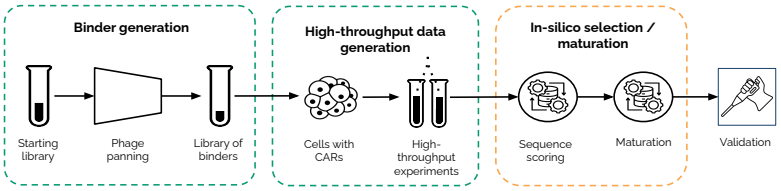


Figure 2: Simplified candidate generation pipeline. Using phage display based on a highly diverse starting library we screen for binders to the target of interest using phage display. Resulting candidates are evaluated in cells in scalable proprietary assays.

VHHs consist of three hypervariable complementary determining regions (CDRs) interspersed with framework regions (FRs), as depicted in Figure 1. CDRs play a critical role in antigen binding, with their flexible structures adapting to fit the binding site. In contrast, FRs serve as a scaffold to support the CDRs for effective antigen binding, and therefore exhibit significantly less sequence diversity in comparison to CDRs. CDR3 is the most diverse region and has a longer length distribution in VHHs than in scFvs, allowing for more versatility in targeting antigen binding sites.

For CAR T-cell therapy to be efficacious, the designed protein must fulfil several criteria. The protein must be successfully transduced into T-cells via a suitable vector, express well, bind specifically to the target antigen, and finally trigger T-cell activation, ideally in a target-dependent manner. This complex behaviour makes the discovery or design of CARs a challenging problem.

## 3 Hit maturation of CARs using ML

In this work we explore Protein Language Models (PLMs) as a potential avenue to enable hit maturation in CARs - efficient, ML-guided exploration of the vast design space, going beyond naive approaches such as deep mutational scans [Fowler and Fields, 2014] or substitution methods [Henikoff and Henikoff, 1992]. We focus on the performance of CARs against a variety of targets, to discover patterns that either enable good performance (such as activation, expression, and specificity) or that negatively impact the cell.

We specifically use auto-regressive models in this work, as they i) allow us to assess the likelihood of a sequence under the model, ii) allow efficient sampling of the distribution learned by the model in order to explore mutations around an initial candidate[1], iii) have well-established methods and tools for fine-tuning after pre-training, and iv) may enable zero-shot approaches to the design of CARs.

### 3.1 High-throughput evaluation of CARs

To align PLMs to the task of hit maturation, we need to derive sufficient experimental data about the performance of CARs. The goal is to determine the suitability of a CAR for a specific target and to assign a scalar score that is well correlated with CAR performance.

Our platform, as depicted in Figure 2, starts of with a semi-synthetic VHH library, wherein most of the diversity ($\sim 10^{11}$) exists in the CDR3 region. We first isolate binders for a specific target by creating a VHH phage display library. These phages are exposed to immobilised target protein and non-specific binders are washed away in a process known as phage panning. We conduct multiple rounds of this selection process to enrich the population of binders, and use Next Generation Sequencing (NGS) to identify the sequences of the binding population at each step.

The VHHs that bind to the target of interest are subsequently reformatted into a CAR library ($\sim 10^5$). We then generate a pool of CAR-T cells, with each cell expressing a single CAR design. By co-culturing the library-expressing cells with target-expressing cells and target-negative cells, we can assess the level of cell activation and use a fluorescence-activated cell sorter to isolate the activated and non-activated subsets of these libraries. Using NGS we can identify the candidates in each subset. The fractions of activated and non-activated cells for each CAR across different cell lines are used to score their performance, leading to a single scalar associated with each CAR.

[1]Recently, Hayes et al. [2024] have demonstrated how masking LMs can be fine-tuned using similar approaches and how issues with sampling can be circumvented, which we are keen to explore in future work.

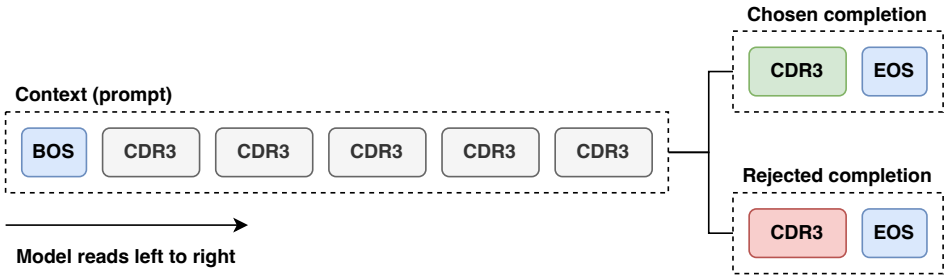


Figure 3: Encoding of a pair of chosen and rejected completion for a context prompt. Context- and chosen CDR3s are sampled from good performers for a specific target, rejected CDR3s from poor performers for the same target. We produce up to $n = 10$ different pairs for each good CDR3.

### 3.2 Preference-based fine-tuning of PLMs

In natural language, it is often challenging to assign absolute quality ratings to responses. Human raters can, however, express preferences between two outputs, establishing a partial order. *Preference Optimization* has been developed to fine-tune large language models (LLMs) based on this signal, and techniques such as Reinforcement Learning with Human Feedback (RLHF) [Christiano et al., 2017] have become the most common approach for the alignment of LLMs. Our setting is similar – high-throughput testing of CARs, while scalable, may lack sufficient precision to resolve the finest difference between candidate CARs. In line with other recent work in the field [Hayes et al., 2024] we therefore formulate a preference-optimisation task.

We use *Direct Preference Optimisation* (DPO) [Rafailov et al., 2024] to fine-tune models based on data derived from high-throughput experiments. DPO is a simplified approach to preference optimisation, which does not require a separate reward model, but instead formulates a simple classification problem to improve the probability that the chosen response is preferred over the rejected response. More formally, DPO relies on a sigmoid-based learning objective:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta \mid \pi_{\text{st}}, \mathcal{D}_p) = \mathbb{E}_{(x,y_w,y_u)\sim\mathcal{D}_p}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{st}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_u \mid x)}{\pi_{\text{st}}(y_u \mid x)}\right)\right] \tag{1}$$

where $\pi_{\text{st}}$ is the pretrained model, $\pi_\theta$ the model being fine-tuned, $x$ is the context (prompt), and $y_w, y_u$ are the chosen and rejected completions. The idea behind this loss is to maximise the margin between the rejected and chosen completions conditioned on the same context. We explore other loss functions beyond the sigmoid loss in equation 1:

**Hinge,** as proposed by [Liu et al., 2024] and originally inspired by [Zhao et al., 2023];

$$\mathcal{L}_{\text{hinge}}(\pi_\theta \mid \pi_{\text{st}}, \mathcal{D}_p) = \mathbb{E}_{(x,y_w,y_u)\sim\mathcal{D}_p}\left[\max\left(0, 1 - \left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\text{st}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_u \mid x)}{\pi_{\text{st}}(y_u \mid x)}\right)\right)\right] \tag{2}$$

**Kahneman-Tversky Optimization** [Ethayarajh et al., 2024]

$$\mathcal{L}_{\text{KTO}}(\pi_\theta \mid \pi_{\text{st}}, \mathcal{D}_p) = \mathbb{E}_{(x,y_w,y_u)\sim\mathcal{D}_p}\left[1 - v(x,y)\right] \tag{3}$$

where

$$v(x,y) = \begin{cases} \sigma\left(\beta\left(\log\frac{\pi_\theta(y|x)}{\pi_{st}(y|x)} - \text{KL}(\pi_\theta(y \mid x)\|\pi_{st}(y \mid x))\right)\right) & \text{if } y = y_{\text{w}} \\ \sigma\left(\beta\left(\text{KL}(\pi_\theta(y \mid x)\|\pi_{st}(y \mid x)) - \log\frac{\pi_\theta(y|x)}{\pi_{st}(y|x)}\right)\right) & \text{if } y = y_{\text{u}} \end{cases}$$

We set $\lambda_W$ and $\lambda_U$ introduced by [Ethayarajh et al., 2024] to 1 to weigh chosen and rejected completions equally, and omit them in equation 3.

### 3.3 Preference dataset

We construct a preference dataset $\mathcal{D}_p$ based on scored candidates across a number of different target antigens. We group all candidate sequences by CDR3 (retaining the maximum performing variant of

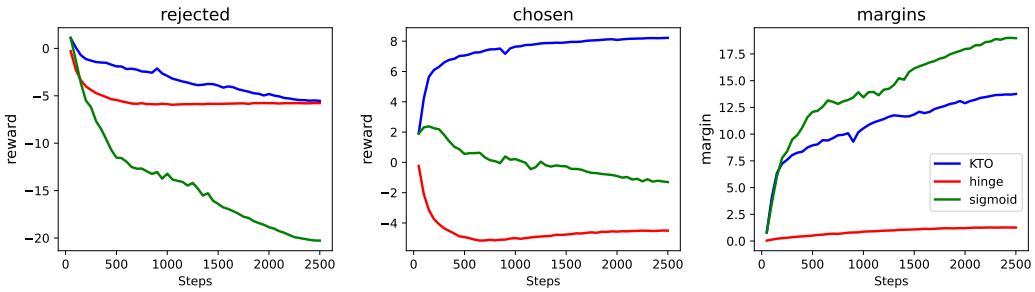


Figure 4: Comparison between rewards, the mean difference between the log probabilities of $\pi_\theta$ and $\pi_{\text{st}}$ for rejected and chosen completions, and margins between them, for models trained using the three analysed loss functions (best performing hyperparameters were chosen for each loss function). Models trained using the original sigmoid-based loss penalise the rejected completions heavily, and tend to become overly confident in their decisions, making them more susceptible to overfitting and numerical instability. They could perhaps benefit from more sophisticated regularisation mechanisms. Models trained with hinge loss maintain very small margins, which may explain their tendency to produce trivial completions, despite promising validation loss and accuracies. It is possible that in the future iterations, after construction of datasets with harder examples [Robinson et al., 2021], models using this loss have some potential to yield useful results due to their rapid convergence and stability. KTO's behaviour appears to be best aligned with our desired use-case of hit maturation, as it increases the likelihood of the chosen completion instead of penalising the rejected completion, all while achieving high margins.

other CDRs if relevant) to form the set of CDR3s $S$ per target antigen, and their associated scores $P$, and split them into good performers (score larger than $t_c$) and poor performers (score less than $t_r$). For every good performer, we construct up to $n = 10$ preference pairs by randomly sampling a poor performer and concatenating $k = 5$ randomly sampled good performers as context (prompt). Intuitively, these context CDR3s softly encode the target antigen and should allow the model to tailor its conditional distribution to the target during fine-tuning, potentially enabling generalisation to new targets that were not present in the training set (even though this is not the focus of this work).

The difficulty of the task can be controlled via the threshold parameters $t_c$ and $t_r$. Using this procedure and conservative thresholds we construct a total of $70k$ pairs with $3k$ pairs held out as validation set with a fully disjoint set of context and candidate CDR3s. Candidate sequences are restricted to CDR3s and have an average length between 10 and 11 amino acids. An example preference pair is illustrated in Figure 3.

### 3.4 Fine-tuning *ProGen2* based on preference data

Throughout this work we rely on *ProGen2* [Nijkamp et al., 2023] as the pretrained model due to the publicly available implementation and wide choice of pretrained weights, ranging from 151M to 6.4B parameters. Pretrained on approximately 280M protein sequences, ProGen2 serves as a robust baseline for fine-tuning. We do not apply traditional supervised fine-tuning prior to DPO-based training, as we did not observe a tangible benefit in preliminary experiments.

Our implementation is based on Huggingface's DPO trainer, adapting configurations based on model size and hardware constraints. We explore small (151M), medium (764M) and large (2.7B) model variants. Experiments are conducted on Google's Cloud Platform, using 8 NVIDIA A100 GPUs (80GB) per machine for large models, 4 for medium models; and a single NVIDIA T4 for small models.

We explore learning rates from $10^{-4}$ to $10^{-8}$ and a linear learning rate decay. We use per-device batch sizes of 16 for large models, 32 for medium models, 32 for small models, and gradient accumulation to achieve larger effective batch-sizes. We tune $\beta$, which controls deviation from the base model, and find $\beta = 0.1$ works well across most experiments. We train models for approximately 10 epochs, which we found sufficient to ensure convergence, and retain the checkpoint with the lowest validation loss for further experiments.

### 3.5 Model selection

We found the medium-sized variant of *Progen2* to be the best trade-off between performance and resource efficiency, where large models did not show improved performance while requiring significantly more compute.

Overall, various combinations of hyper-parameters showed promising training metrics, where we observed validation accuracy beyond 75%. However, manual inspection revealed that some fine-tuned models produced trivial or collapsed CDR3 sequences when greedily sampling from the model conditioned on a suitable context, which was not reflected in the training metrics. Notably, models trained with hinge loss often overfitted quickly and performed well on the preference dataset, yet generated trivial outputs. We speculate that models trained with KTO loss prioritise the utility of the generated sequences over simply maximising the log-likelihood of preferences, as argued by [Ethayarajh et al., 2024]. This is illustrated in Figure 4.

The remaining evaluation in this paper is thus focused on a medium-sized model (764M parameters) trained with the KTO loss. Training and validation metrics for various learning rates are illustrated in Figure 8 in the appendix.

## 4 Evaluation

We aim to evaluate whether a fine-tuned model can be successfully applied to hit maturation, where we explore the design space surrounding a candidate CAR discovered in the pipeline outlined above. Overall we pursue two questions: i) Can we exceed the performance of the candidate CAR by exploring single or double mutants, and ii) is model loss correlated to the performance of CARs in a way that would enable guided exploration of the CAR design space around selected candidates?

We conduct assays in 96-well plates, where each well contains cells expressing a single synthesised CAR with up to two amino acid mutations. This enables us to assess the immune response (activation) individually for each CAR, replicating the best-practice (low throughput) evaluation. We conduct three experiments:

**Greedy generation** for two manually selected candidate CARs that showed promising performance in previous experiments for a target included in the training set, we construct a suitable context of 5 well performing CDR3s, and generate candidates greedily left-to-right, following suggested top 3 substitutions for each position by the model as outlined in algorithm 1. Based on likelihood (model loss[2]), we pick the top 15 single mutants, and the top 15 double mutants to evaluate on a 96-well plate. Two of the selected mutants were in the training-set, and have been removed from further analysis below.

**Exhaustive search** For the same two candidates, we score all possible single and double mutants using the model across all possible permutations of 5 context CDR3s to get the average likelihood, and pick the top scoring 45 mutants for each candidate to evaluate in a 96-well plate. Three of the selected mutants were in the training-set, and have been removed from further analysis below.

**Few shot maturation** To replicate the desired use-case for hit maturation we select 10 additional candidate CARs (of which 7 were in the training set). For each CAR, we do an exhaustive search for single and double mutants as outlined above, and evaluate the top 8 mutants along with each candidate CAR on a 96-well plate. None of the selected mutants were part of the training set.

Note that exhaustive search of all single and double mutants and scoring them across all 120 context permutations is computationally significantly more expensive than the alternative greedy approach. Intuitively, exhaustive search may find candidates with higher likelihood than the greedy approach, as each position is only conditioned on whatever is "left" of the position, even if a different substitution would significantly increase the likelihood of the remaining positions to the "right".

### 4.1 Results

Figure 5 illustrates all single and double mutants discovered using the greedy approach along with a performance score of T cell reporter activation ($\Delta$GFP). Overall, we observe a strong Pearson

[2]We rely on cross-entropy for next-token prediction as a surrogate for likelihood

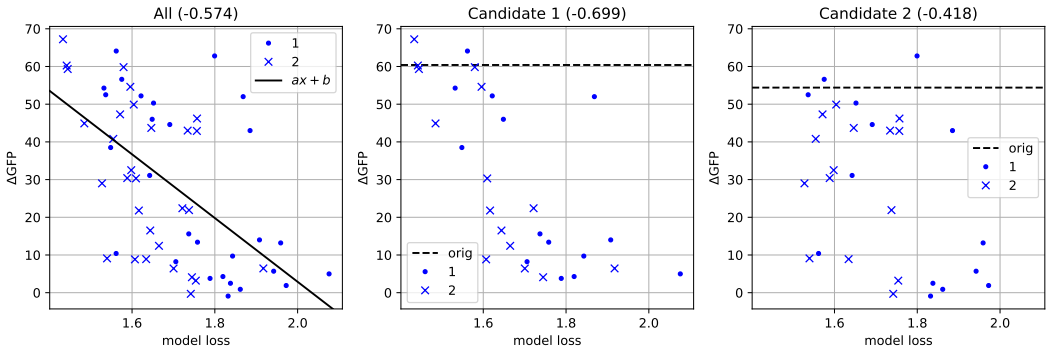


Figure 5: Model loss vs activation for greedily generated candidates. All plots show model loss averaged over all possible context permutations, plotted against activation measured as ΔGFP. Dots indicate candidates with a single mutation, crosses two mutations, black lines indicate baseline performance of each candidate. Overall we see strong correlation between averaged model loss and activation.

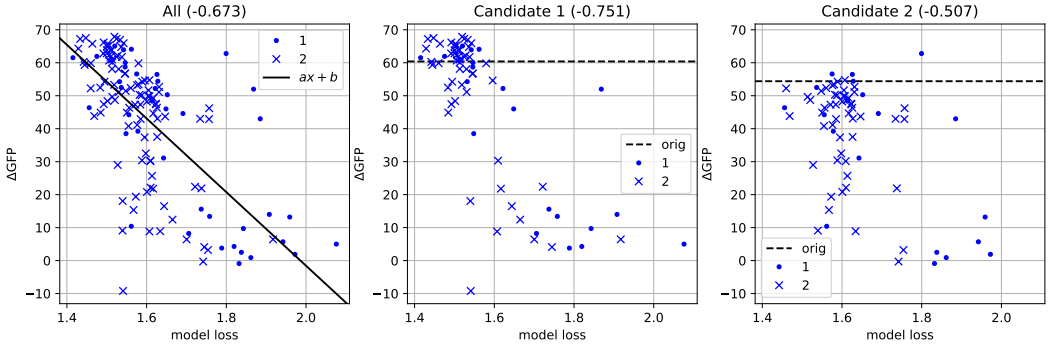


Figure 6: Model loss vs activation for greedily generated candidates and candidates from exhaustive search. All plots show model loss averaged over all possible context permutations, plotted against activation measured as ΔGFP. Dots indicate candidates with a single mutation, crosses two mutations, black lines indicate baseline performance of each candidate.

correlation between average model loss and ΔGFP ($-0.574$). This correlation is higher for Candidate 1 ($-0.699$), but in all cases correlation is significant at $p < 0.05$. We also assess the correlation of model loss to ΔGFP of the pretrained model before fine-tuning (see Figure 9 in the Appendix), and find that fine-tuning significantly improves correlation for both candidate 1 ($-0.699$ vs. $0.218$) and candidate 2 ($-0.418$ vs. $0.072$).

Figure 6 shows the performance for mutants discovered greedily along with those discovered via exhaustive search. To account for systematic bias, we normalise ΔGFP on the second plate using the performance of each reference CAR as a guide. We see, perhaps surprisingly, that the majority of top candidates found via exhaustive search show promising performance and cluster tightly around the performance of each baseline. In particular, we discover many single and double mutants of candidate 1 that significantly outperform the baseline. Both greedy as well as exhaustive search yield better correlation and more favourable performance for candidate 1, even though both candidates were in the training set (see figure 5 and 6).

Figure 7 shows the performance of mutants relative to their parent for seven candidates from the training set (S1-S7) and three candidates from the validation set (S8-S10). In the majority of cases we discover at least a few mutants that outperform their parent, and in three cases mutants that show more than double the performance – a promising result given the size of the search space ($10^4$ - $10^5$). Performance further appears to generalise to the validation set (S8-S10), but we do not yet have sufficient experimental data to determine statistical significance.

Further work is necessary to explore the factors that influence the performance of the approach, to determine if the source of the discrepancy between candidate 1 and candidate 2 is biological (i.e.

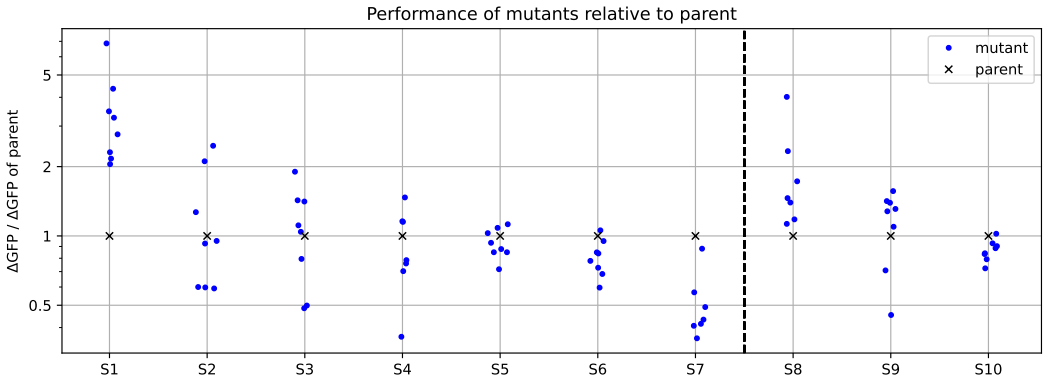


Figure 7: For 10 selected sequences, 8 mutants are evaluated in a 96-well plate. The parents in S1-S7 are from the training set, and S8-S10 are from the validation set. None of the mutants occur in the preference dataset.

**Algorithm 1** Greedy diversification

**function** GREEDYGEN($L, R, k$, topk)
  $D$ = [concat$(L, R)$]
  **if** $k$ == 0 or $|R| < 2$ **then**
    return $D$
  **end if**
  S = [$\text{argmax}_{\text{topk}}(\text{logits}(\text{concat}(L, R))_p)$ $\forall p \in [1 \ldots (|R|-1)]$]  ▷ get top k subst. per position in $R$
  **for** $p \in [1 \ldots (|R|-1)]$ **do**
    **for** $s \in S_p$ **do**
      **if** valid$(s)$ **then**
        D = $D \cup$ GREEDYGEN(concat$(L, R_{:p}, s), R_{(p+1):}, k-1$, topk)
      **end if**
    **end for**
  **end for**
  return$(D)$
**end function**

structural), or whether it may be an artefact from the construction of the preference dataset. While these preliminary results are difficult to interpret due to low number of candidates, we can draw two conclusions: i) Overall model loss seems to be well correlated to candidate performance, which supports our hypothesis that such models can guide exploration of the design space surrounding a candidate CAR, and ii) to maximise the chances of discovering high-performing mutants, it is crucial to conduct an exhaustive search, even if this comes at higher computational cost. Recent advances in sampling from PLMs [Hayes et al., 2024], may address this limitation, which we leave for future work to explore.

### 4.2 Discussion

In this work we explore Protein Language Models (PLMs) for hit maturation in cell therapy, and demonstrate how data from a high-throughput experimental platform can be harnessed to formulate a preference task. Even with relatively simple experiments, using well-established methods and pre-trained publicly available models, we obtain promising results – highlighting the significant potential of PLMs in cell therapy, which could generalise to other therapeutic modalities.

As the presented results are preliminary there remains much to explore, including leveraging multi-dimensional single-cell data to capture more comprehensive information on gene expression profiles, which would potentially alleviate the lack of precision of high-throughput testing. Furthermore, our approach is agnostic to spatial protein structure, where Graph Learning techniques [Hetzel et al., 2021, Jiang et al., 2024] could lead to improved performance and better understanding of which candidates have the most potential for maturation.

## References


J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, O. Ronneberger, L. Willmore, A. J. Ballard, J. Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pages 1–3, 2024.

S. Bachas, G. Rakocevic, D. Spencer, A. V. Sastry, R. Haile, J. M. Sutton, G. Kasun, A. Stachyra, J. M. Gutierrez, E. Yassine, et al. Antibody optimization enabled by artificial intelligence predictions of binding affinity and naturalness. *BioRxiv*, pages 2022–08, 2022.

U. Blache, S. Tretbar, U. Koehl, D. Mougiakakos, and S. Fricke. Car t cells for treating autoimmune diseases. *RMD open*, 9(4):e002907, 2023.

K. M. Cappell and J. N. Kochenderfer. Long-term outcomes following car t cell therapy: what we know so far. *Nature reviews Clinical oncology*, 20(6):359–371, 2023.

R. Castellanos-Rueda, R. B. Di Roberto, F. Bieberich, F. S. Schlatter, D. Palianina, O. T. Nguyen, E. Kapetanovic, H. Läubli, A. Hierlemann, N. Khanna, et al. speedingcars: accelerating the engineering of car t cells by signaling domain shuffling and single-cell sequencing. *Nature Communications*, 13(1):6555, 2022.

P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei. Deep reinforcement learning from human preferences. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 4299–4307. Curran Associates, Inc., 2017.

T. Clark, V. Subramanian, A. Jayaraman, E. Fitzpatrick, R. Gopal, N. Pentakota, T. Rurak, S. Anand, A. Viglione, R. Raman, et al. Enhancing antibody affinity through experimental sampling of non-deleterious cdr mutations predicted by machine learning. *Communications Chemistry*, 6(1): 244, 2023.

G. Corso, H. Stärk, B. Jing, R. Barzilay, and T. Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

K. G. Daniels, S. Wang, M. S. Simic, H. K. Bhargava, S. Capponi, Y. Tonai, W. Yu, S. Bianco, and W. A. Lim. Decoding car t cell phenotype using combinatorial signaling motif libraries and machine learning. *Science*, 378(6625):1194–1200, 2022.

K. Ethayarajh, W. Xu, N. Muennighoff, D. Jurafsky, and D. Kiela. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*, 2024.

D. M. Fowler and S. Fields. Deep mutational scanning: a new style of protein science. *Nature methods*, 11(8):801–807, 2014.

B. Gorovits and E. Koren. Immunogenicity of chimeric antigen receptor t-cell therapeutics. *BioDrugs*, 33(3):275–284, 2019.

T. Hayes, R. Rao, H. Akin, N. J. Sofroniew, D. Oktay, Z. Lin, R. Verkuil, V. Q. Tran, J. Deaton, M. Wiggert, et al. Simulating 500 million years of evolution with a language model. *bioRxiv*, pages 2024–07, 2024.

Y. He, X. Zhou, C. Chang, G. Chen, W. Liu, G. Li, X. Fan, M. Sun, C. Miao, Q. Huang, et al. Protein language models-assisted optimization of a uracil-n-glycosylase variant enables programmable t-to-g and t-to-c base editing. *Molecular Cell*, 84(7):1257–1270, 2024.

S. Henikoff and J. G. Henikoff. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

L. Hetzel, D. S. Fischer, S. Günnemann, and F. J. Theis. Graph representation learning for single-cell biology. *Current Opinion in Systems Biology*, 28:100347, 2021.

B. L. Hie, V. R. Shanker, D. Xu, T. U. Bruun, P. A. Weidenbacher, S. Tang, W. Wu, J. E. Pak, and P. S. Kim. Efficient evolution of human antibodies from general protein language models. *Nature Biotechnology*, 42(2):275–283, 2024.

F. Jiang, Y. Guo, H. Ma, S. Na, W. Zhong, Y. Han, T. Wang, and J. Huang. Gte: a graph learning framework for prediction of t-cell receptors and epitopes binding specificity. *Briefings in Bioinformatics*, 25(4):bbae343, 2024. doi: 10.1093/bib/bbae343.

S. Joubbi, A. Micheli, P. Milazzo, G. Maccari, G. Ciano, D. Cardamone, and D. Medini. Antibody design using deep learning: from sequence and structure design to affinity maturation. *Briefings in Bioinformatics*, 25(4), 2024.

Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

T. Liu, Y. Zhao, R. Joshi, M. Khalman, M. Saleh, P. J. Liu, and J. Liu. Statistical rejection sampling improves preference optimization. In *International Conference on Learning Representations (ICLR)*, 2024.

A. Mullard. Parsing clinical success rates. *Nature Reviews Drug Discovery*, 15(7):447–448, 2016.

M. Mullin, J. McClory, W. Haynes, J. Grace, N. Robertson, and G. van Heeke. Applications and challenges in designing vhh-based bispecific antibodies: leveraging machine learning solutions. In *Mabs*, volume 16, page 2341443. Taylor & Francis, 2024.

S. Muyldermans. Nanobodies: natural single-domain antibodies. *Annual review of biochemistry*, 82 (1):775–797, 2013.

E. Nijkamp, J. A. Ruffolo, E. N. Weinstein, N. Naik, and A. Madani. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

T. H. Olsen, F. Boyles, and C. 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.

D. Prihoda, M. Mullin, W. Haynes, J. Grace, and G. van Heeke. Biophysical characterization of antibody variants to determine developability. In *Biophysical Journal*, volume 121, page 635a. Elsevier, 2022.

R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

J. Robinson, C.-Y. Chuang, S. Sra, and S. Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*, 2021. URL `https://doi.org/10.48550/arXiv.2010.04592`.

J. A. Ruffolo, J. J. Gray, and J. Sulam. Deciphering antibody affinity maturation with language models and weakly supervised learning. *arXiv preprint arXiv:2112.07782*, 2021.

R. W. Shuai, J. A. Ruffolo, and J. J. Gray. Generative language modeling for antibody design. *BioRxiv*, pages 2021–12, 2021.

M. Steinegger and J. Söding. Clustering huge protein sequence sets in linear time. *Nature communications*, 9(1):2542, 2018.

B. E. Suzek, Y. Wang, H. Huang, P. B. McGarvey, C. H. Wu, and U. Consortium. Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6):926–932, 2015.

I. H. M. Tobias H. Olsen and C. M. Deane. Ablang: An antibody language model for completing antibody sequences. *bioRxiv*, 2022. doi: https://doi.org/10.1101/2022.01.20.477061.

A. Vaswani. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

J. L. Watson, D. Juergens, N. R. Bennett, B. L. Trippe, J. Yim, H. E. Eisenach, W. Ahern, A. J. Borst, R. J. Ragotte, L. F. Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Y. Zhao, R. Joshi, T. Liu, M. Khalman, M. Saleh, and P. J. Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023. doi: 10.48550/arXiv.2305.10425.

# A Appendix

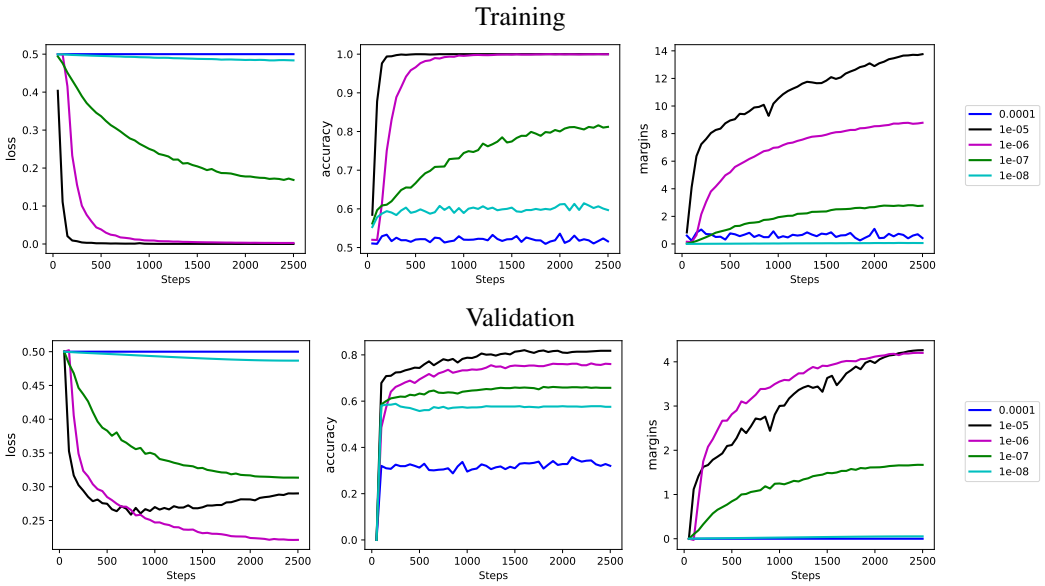


Figure 8: Training (top row) and validation metrics (bottom row) for various settings of learning rate in a *progen2-medium* trained with KTO. Loss remains high for both the smallest and largest learning rates. Based on these results and further qualitative evaluation (see text for details), we selected $10^{-5}$ for further experimentation in this work.

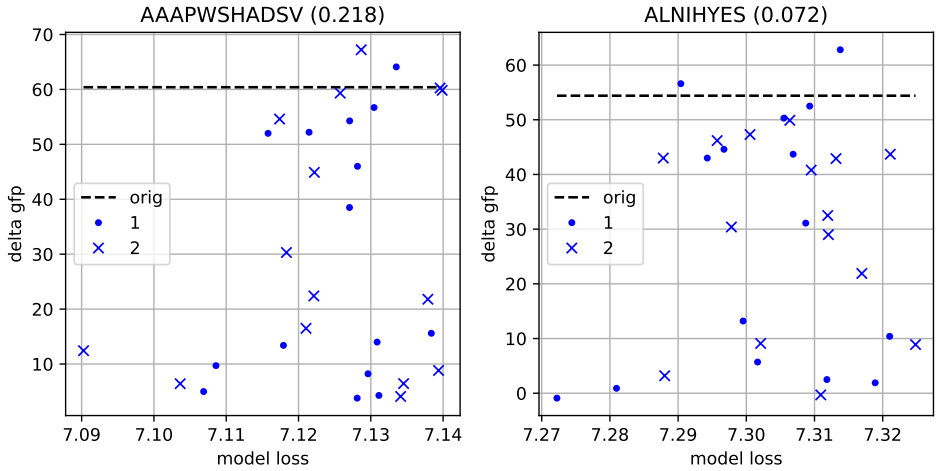


Figure 9: Model loss of *progen2-medium* before fine-tuning vs activation for greedily generated candidates. All plots show model loss averaged over all possible context permutations, plotted against activation measured as $\Delta$GFP. Dots indicate candidates with a single mutation, crosses two mutations, black lines indicate baseline performance of each candidate. As expected, performance before fine-tuning is poor (insignificant Pearson correlation of $0.218$ and $0.072$), where some of the worst performing mutants obtain the lowest loss. Without fine-tuning, this model would not be helpful when exploring the design space of CARs, motivating the need for preference-based fine tuning of PLMs.