MAMMAL - Molecular Aligned Multi-Modal Architecture and Language

Yoel Shoshan[1*†], Moshiko Raboh[1†], Michal Ozery-Flato[1†], Vadim Ratner[1], Alex Golts[1], Jeffrey K. Weber[2], Ella Barkan[1], Simona Rabinovici-Cohen[1], Sagi Polaczek[1], Ido Amos[1], Ben Shapira[1], Liam Hazan[1], Matan Ninio[1], Sivan Ravid[1], Michael M. Danziger[1], Joseph A. Morrone[2], Parthasarathy Suryanarayanan[2], Michal Rosen-Zvi[1], Efrat Hexter[1]

[1]IBM Research Labs, IBM Research, Haifa, 3498825, Israel.
[2]IBM TJ Watson Research Center, IBM Research, 1101 Kitchawan Rd., NY, 10598, Yorktown Heights, USA.

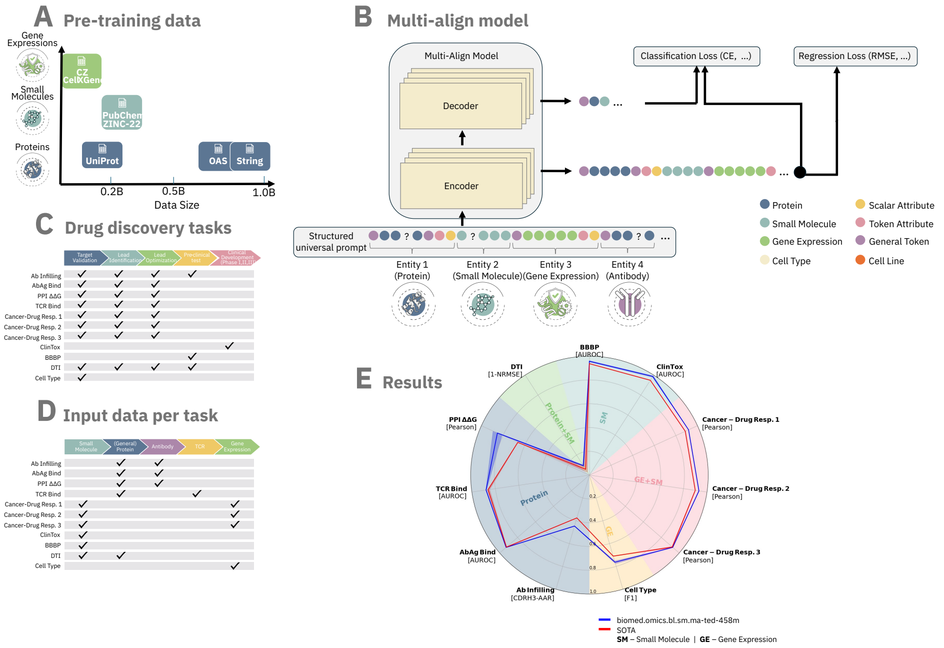
*Corresponding author(s). E-mail(s): yoels@il.ibm.com;
†These authors contributed equally to this work.

Abstract

Drug discovery typically consists of multiple steps, including identifying a target protein key to a disease's etiology, validating that interacting with this target could prevent symptoms or cure the disease, discovering a small molecule or biologic therapeutic to interact with it, and optimizing the candidate molecule through a complex landscape of required properties. Drug discovery related tasks often involve prediction and generation while **considering multiple entities that potentially interact**, which poses a challenge for typical AI models. For this purpose we present **MAMMAL** - **M**olecular **A**ligned **M**ulti-**M**odal **A**rchitecture and **L**anguage - a method that we applied to create a versatile multi-task foundation model `ibm/biomed.omics.bl.sm.ma-ted-458m` that learns from large-scale biological datasets (2 billion samples) across diverse modalities, including proteins, small molecules, and genes. We introduce a prompt syntax that supports a wide range of classification, regression, and generation tasks. It allows combining different modalities and entity types as inputs and/or outputs. Our model handles combinations of tokens and scalars and enables the generation of small molecules and proteins, property prediction, and transcriptomic lab test predictions. We evaluated the model on 11 diverse downstream tasks spanning different steps within a typical drug discovery pipeline, where it reaches

new SOTA in 9 tasks and is comparable to SOTA in 2 tasks. **This performance is achieved while using a unified architecture serving all tasks, in contrast to the original SOTA performance achieved using tailored architectures.**

The model code and pretrained weights are publicly available at https://github.com/BiomedSciAI/biomed-multi-alignment and https://huggingface.co/ibm/biomed.omics.bl.sm.ma-ted-458m.



	Target Validation	Lead Identification	Lead Optimization	Pre-clinical test	Clinical Development (Phase I,II,III)
Ab Infilling	✓	✓	✓	✓	
AbAg Bind	✓	✓	✓		
PPI ΔΔG	✓	✓	✓		
TCR Bind	✓	✓	✓		
Cancer-Drug Resp. 1	✓	✓	✓		
Cancer-Drug Resp. 2	✓	✓	✓		
Cancer-Drug Resp. 3	✓	✓	✓		
ClinTox					✓
BBBP				✓	
DTI	✓	✓	✓	✓	
Cell Type	✓				

	Small Molecule	(General) Protein	Antibody	TCR	Gene Expression
Ab Infilling		✓	✓		
AbAg Bind		✓	✓		
PPI ΔΔG		✓	✓		
TCR Bind		✓		✓	
Cancer-Drug Resp. 1	✓				✓
Cancer-Drug Resp. 2	✓				✓
Cancer-Drug Resp. 3	✓				✓
ClinTox	✓				
BBBP	✓				
DTI	✓	✓			
Cell Type					✓

Fig. 1 **(A)** We introduce a multi-aligned model pretrained on six datasets, each containing tens to hundreds of millions of data points. These data points include protein sequences, small molecules, and gene expression profiles, with a combined sample size of 2 billion. **(B)** The multi-aligned model combines flexible encoder-only and encoder-decoder components. It takes sequences as input, which may contain any combination of tokens and scalar elements, processed by an encoder stack consisting of self-attention blocks. In encoder-only mode, a dedicated token prediction head outputs logits for token predictions, with an optional scalar prediction head for scalar outputs. In encoder-decoder mode, residual connections inject features from the encoder's final hidden layer into each decoder layer, and a decoder-specific prediction head outputs the final logits. **(C)** Diverse downstream tasks performed by the multi-aligned model, mapped to their contributions within the steps of a typical drug discovery pipeline. **(D)** Diverse downstream tasks performed by the multi-aligned model, categorized by data type used in the fine-tuning process. **(E)** Performance of the multi-aligned model across a diverse set of tasks compared to SOTA.

1 Introduction

Drug discovery traditionally follows a multi-step pipeline that begins with identifying disease-associated proteins, progresses to finding compounds that can effectively target these proteins, and culminates in the optimization of drug candidates to meet rigorous standards for efficacy and safety. This process is both costly and labor-intensive [1], requiring extensive laboratory assays that measure drug-target interactions, assess cellular changes in disease-relevant cell lines, and validate therapeutic efficacy and safety [1**?** –3]. Drugs in development can be either small molecules, which are stable, easy to manufacture, and suitable for oral delivery [4], or biologic therapeutics, such as engineered antibodies, which offer high specificity but require complex manufacturing and are typically administered by injection [5].

Accelerating drug discovery has become a central focus in biomedical research, aiming to streamline target identification, drug design, and testing [6–8]. Analyzing gene expression profiles, particularly from single-cell RNA sequencing (RNA-seq), has emerged as a key tool for distinguishing between cell populations associated with different diseases [9–12]. This analysis enhances our understanding of disease mechanisms, facilitates the identification of new drug targets, and allows for the examination of drug effects across various cell types [13, 14]. In drug design, trained generative models are utilized to synthesize new drug candidates for further exploration [15–17]. As high-throughput screening assays for measuring drug binding affinity are costly and challenging to scale, accurately predicting drug-target interactions can significantly enhance drug design, improving both efficacy and precision. Overall, predictive modeling of binding affinity, toxicity, and efficacy in the early stages of the pipeline can reduce reliance on expensive late-stage testing, ultimately saving time and resources in drug development.

When creating predictive and generative AI models, one of the key challenges in the field involves the question - how should different modalities and entities be combined as inputs/outputs to a model ?[18] This uncertainty is especially dominant in predictive and generative tasks that involve interaction between entities, for example, predicting whether a specific antibody and a protein target are likely to bind or not.

In this work, we introduce the MAMMAL (Molecular Aligned Multi-Modal Architecture and Language) method, and develop a multi-aligned foundation model paired with a prompt syntax that integrates multiple data domains to support a wide range of drug discovery tasks. The multi-aligned model, which enables aligning multiple entities into a single prompt, has been extensively pre-trained on 2 billion samples from diverse datasets, using auxiliary tasks of mask infilling, denoising, generation, and classification. The multi-aligned model is compatible with both encoder-decoder and encoder-only architectures and effectively incorporates numerical values through continuous token embedding, enhancing numerical precision and reducing vocabulary size. We rigorously evaluate the multi-aligned model across 11 downstream tasks - spanning classification, regression, and generation - covering key stages of the drug discovery pipeline across three primary domains: small molecules, proteins, and gene expression profiles. The multi-aligned model achieves state-of-the-art performance in 9 tasks and matches top performance in the remaining 2 tasks. The model is publicly available on https://huggingface.co/ibm/biomed.omics.bl.sm.ma-ted-458m under the name `ibm/biomed.omics.bl.sm.ma-ted-458m`

2 Methods

The MAMMAL method is built around three core components: the model architecture, the molecular prompt syntax, and extensive pretraining. In Subsection 2.1, we detail the architecture and its enhancements to the standard transformer framework. Subsection 2.2 focuses on the molecular prompt syntax, a key feature that enables the support of a diverse range of pretraining and downstream tasks for drug discovery. Finally, Subsection 2.3 outlines the pretraining process, which fascilitates leveraging large, cross-domain datasets and handling multiple entities simultaneously.

2.1 MAMMAL Architecture

The MAMMAL framework builds on the transformer architecture introduced by Vaswani et al. [19], and is inspired by the T5 framework [20] to formulate tasks as sequence-to-sequence problems within a unified model. MAMMAL introduces three primary features:

- Task modeling in either an encoder-only mode, akin to BERT [21], or an encoder-decoder autoregressive mode [19]. The weights of the encoder stack are shared across both modes, enabling multi-task training that integrates tasks from both types. Model parameter updates are performed through gradient accumulation across all tasks.
- Integral support for the molecular prompt syntax through the new Modular Tokenizer component, which facilitates the extension of molecular domain vocabularies and the incorporation of new domain vocabularies without necessitating the retraining of existing models.
- Supporting numerical values (scalar) as both inputs being fed into the model, and also as outputs that the model can learn to predict. This is done in a continuous way, not requiring any binning or translation to a discrete space.

More details on the architecture and how scalars inputs and outputs are supported can be found in Appendix A

2.2 Prompt Syntax

The prompt syntax employs a nomenclature of special tags that represent elements of molecular entities, molecular sequences, and their attributes, as well as interactions within the broader molecular system. It is designed to accommodate multiple data domains by providing tokenization hints for different segments of the input sequence, with all tokenizers supporting a common set of special tokens, such as ⟨EOS⟩. Numeric values are handled by a designated tokenizer, and the syntax is applicable to both model inputs and outputs. Furthermore, the syntax is extensible, allowing for the addition of new tags to each tokenizer or to the common set of special tokens. The

Table 1 Pretraining Tasks

Name	Domain	Entity Type	Task Type	Dataset	Number of Samples
Protein LM	Biologic	General Protein	Spans Masking LM	Uniref90 [22]	180M
Antibody LM	Biologic	Antibody	Spans Masking LM	OAS [23]	650M
Small Molecule LM	Small Molecules	Small Molecule	Spans Masking LM	ZINC [24] + PubChem [25]	200M
Cell Genes LM	Single Cell Transcript-omics	Cell Genes	Spans Masking LM	CELLxGENE [26]	30M
Protein-Protein Interaction	Biologic	General Protein	Classification	STRING [27]	780M
Protein-Protein Interaction Gen.	Biologic	General protein	Generation	STRING [27]	390M
Antibody Denoise	Biologic	Antibody	Denoise Sequence	OAS [23]	650M

Details on the pretraining tasks that were used while training `ibm/biomed.omics.bl.sm.ma-ted-458m`.
"Number of Samples" describes the post filtering number of samples actually used. A single model was pretrained with all of the listed tasks, accumulating knowledge spanning multiple domains.

modular tokenizer ensures backward compatibility of newly trained models with existing ones, even after the introduction of new tags or domain-specific tokenizers. More details and examples of the prompt syntax can be found in Appendix B.

2.3 Pretraining

MAMMAL is designed as a comprehensive foundation model, capable of spanning multiple domains and accommodating a variety of entities. It is intended to support diverse task types, ranging from representation-focused tasks to generation-oriented ones. To achieve this, MAMMAL is trained on multiple tasks concurrently. Pretraining was conducted on 2 billion samples sourced from six datasets, which are all publicly available, covering three distinct domains across seven tasks. Table 1 summarizes these tasks, detailing the relevant domains, entity types, and specific datasets. Additional details about the pretraining are provided in Appendix C.

2.4 Evaluation

We compiled a comprehensive set of 11 benchmarks covering multiple data domains and task types, including classification, regression and generation, as well as single-entity, multi-entity, and multi-domain tasks. These benchmarks address key stages of the drug discovery process: identifying target cell types (Cell Type) and advancing precision medicine (Cancer-Drug Response 1-3); predicting drug efficacy (BBBP) and safety (ClinTox); predicting the binding affinity of small-molecule drugs to target proteins (DTI); predicting interactions of biological drugs (PPI); and designing new drugs, such as antibodies, to target specific proteins (Ab Infilling). A key criterion in selecting benchmarks was the availability of predefined train, validation, and test splits. For benchmarks with train-validation-test splits, we fine-tuned the model `ibm/biomed.omics.bl.sm.ma-ted-458m` on the training set, selected the best checkpoint based on validation performance, and reported final results on the test set. Unless otherwise noted, standard errors were estimated by training the models with three different random seeds and calculating the standard deviation of their performance on the test set. Detailed descriptions of each benchmark and the fine-tuning methods used to adapt our pre-trained model for these tasks are provided alongside the evaluation results for each benchmark. In one of the benchmarks (DTI), we report performance using the normalized root mean square error (NRMSE), calculated by dividing the root mean square error by the standard deviation of the labels in the test set. We consider our models to outperform the existing SOTA when the improvement in performance, measured by |SOTA − MAMMAL| / SOTA, exceeds 1%.

3 Results

Below, we present each benchmark used to evaluate `ibm/biomed.omics.bl.sm.ma-ted-458m`, along with performance results from the corresponding fine-tuned models. Each benchmark description includes background on the task, its significance for drug discovery, relevant prior models, and data statistics. A summary of the benchmarks, along with SOTA and MAMMAL results, is presented in Table 2 and visualized in Figure 1(E). Examples of encoder inputs and decoder labels for each benchmark are provided in Table S1.

3.1 Cell Type Annotation

Cell type prediction enables researchers to distinguish between different cell populations, such as those associated with various diseases [9–12]. It is also crucial for understanding how diseases or drugs affect different cell types. In recent years, a variety of methods have been developed for this task, including approaches based on marker genes, correlation-based techniques, and annotation using classification [37]. Recent advances in transformer-based and large-scale foundation models [28, 38, 39] have improved performance by utilizing the full list of genes as input, in contrast to earlier methods where gene selection was based on highly Variable genes (HVG).

The input for this task commonly consists of gene expression (GE) values from single-cell RNA-seq data. The benchmark we used is based on the Zheng68k dataset [40], which is derived from human peripheral blood mononuclear cells (PBMCs) and is widely used for evaluating cell-type annotation performance. The dataset contains 68,579 cells across 11 cell types and originally included 32,738 genes. Preprocessing involved normalization, log-transformation of expression values and followed filtering out non-expressed genes, leaving around 20,387 genes. Similar to the approach in [41],

Table 2 Comparison of SOTA and MAMMAL Performance Across Benchmarks

Benchmark	Domain	Type	Metric	SOTA	MAMMAL	Imp.
Cell type	GE	cls	↑ F1	[28] 0.710	0.763±0.012	**7.5** %
BBBP	SM	cls	↑ AUROC	[29] 0.937	0.957±0.006	**2.2** %
ClinTox	SM	cls	↑ AUROC	[29] 0.948	0.986±0.007	**4.0** %
Cancer-Drug Response 1	GE+SM	reg	↑ Pearson	[30] 0.887	0.917±0.001	**3.4** %
Cancer-Drug Response 2	GE+SM	reg	↑ Pearson	[30] 0.900	0.931±0.002	**3.4** %
Cancer-Drug Response 3	GE+SM	reg	↑ Pearson	[31] 0.923 [0.917-0.929]	0.928±0.000	0.5 %
Ab Infilling	Protein	gen	↑ CDRH3-AAR	[32] 0.375	0.446±0.002	**19.0** %
AbAg Bind	Protein	cls	↑ AUROC	[33] 0.924 [0.923-0.925]	0.928±0.002	0.4 %
TCR Bind	Protein	cls	↑ AUROC	[34] 0.862 [0.85-0.868]	0.879±0.003	**2.0** %
PPI $\Delta\Delta G$	Protein	reg	↑ Pearson	[35] 0.663	0.852±0.041	**28.5** %
DTI	Prot.+SM	reg	↓ NRMSE	0.942±0.028 [36]	0.906±0.011	**3.8** %

For NRMSE lower is better. For other metrics (AUROC, CDRH3-AAR, Pearson, Spearman, and F1) higher is better. Each row shows results from a MAMMAL model fine-tuned from `ibm/biomed.omics.bl.sm.ma-ted-458m` for the corresponding task. Abbreviations: in "Type" column: "cls" = classification, "reg" = regression, "gen" = generation. "Imp." = improvement (percentage) of our model over SOTA. In "Domain" column: "GE" = genes expression, "SM" = small molecule, "Prot." = protein.

our model uses a ranked list of expressed gene names, ordered by their expression levels, as input. The label to predict is provided in the cell ontology format "CL:NNNNNN" (see Table S1).

The model `ibm/biomed.omics.bl.sm.ma-ted-458m` was fine-tuned and evaluated using 5-fold cross-validation, while ensuring similar proportions of cell types across the folds. As shown in Tables 2 and S2, MAMMAL outperforms the previous state-of-the-art performance in both accuracy and F1, achieving a 7.5% improvement in F1.

3.2 BBBP and ClinTox

Drugs must meet various criteria regarding both efficacy and safety. In this study, we selected two relevant benchmarks from MoleculeNet [42], a widely used suite of benchmarks for small-molecule drugs: BBBP and ClinTox. The task in the BBBP benchmark is to predict the ability of drugs to penetrate the blood-brain barrier, a crucial factor in the development of drugs targeting the central nervous system. The ClinTox benchmark involves two related tasks: (1) predicting failure in clinical toxicity trials, and (2) predicting FDA approval status. The overall performance on ClinTox is reported as the average performance across these two tasks.

MoLFormer [29], a well-established model for molecular embeddings trained on 1.1 billion SMILES sequences, has achieved state-of-the-art performance on both the BBBP and ClinTox benchmarks. In our study, we adopted the benchmarks from [29], which provided predefined splits for training, validation, and testing. As shown in 2, MAMMAL surpassed MoLFormer on both benchmarks, achieving an average AUROC of 0.937 on BBBP and 0.986 on ClinTox - representing improvements of 2.2% and 4%, respectively, over the state of the art.

3.3 Cancer-Drug Response (CDR)

Identifying drug response at the cellular level is a critical step in the development of new drugs. Two key public databases supporting this effort, particularly in cancer drug development, are the Cancer Cell Line Encyclopedia (CCLE) [43] and the Genomics of Drug Sensitivity in Cancer (GDSC) [44]. CCLE provides multi-omics profiles for around 1,000 cancer cell lines, while GDSC offers data on the drug responses of these lines to hundreds of drugs, commonly measured using the half-maximal inhibitory concentration (IC_{50}). Notable computational models addressed this task [31, 45, 46].

For our study, we used three subsets of the GDSC database: GDSC1 and GDSC2, available through the Therapeutics Data Commons (TDC) [47], and referred in the paper as Cancer-Drug Response 1 and Cancer-Drug Response 2 respectively; and a subset published in [31], referred as Cancer-Drug Response 3. Table S3 summarizes the number of cell lines, drugs, and cell-drug pairs in these datasets. We used the random splits provided by TDC for Cancer-Drug Response 1 and 2, while for Cancer-Drug Response 3, we followed the split methodology outlined in [31], reserving 5% of the data for the test set, stratified by TCGA [48] pathways associated with the cancer cell lines.

During fine-tuning we used only gene-expression profiles and SMILES representations of drugs, as shown in the example query in Table S1. Similar to the input format for cell type annotation, gene-expression profiles were provided as ranked lists of gene names based on their expression levels. For predicting continuous IC_{50} values, MAMMAL was utilized in regression mode, taking advantage of its built-in support for floating point scalar predictions. As demonstrated in Table 2, our model outperforms the current SOTA models for Cancer-Drug Response 1 and 2, achieving a 3.4% increase in Pearson correlation values. Additionally, it yields results comparable to the SOTA for the Cancer-Drug Response 3 benchmark, with a slight improvement of 0.5%..

3.4 Targeted Antibody Design

Antibodies are a family of proteins produced by the immune system to neutralize foreign antigens and are of particular interest due to their high specificity and strong binding to target molecules [49, 50]. These characteristics have made them a crucial class of therapeutics, driving significant research efforts into the design of new antibody-based drug candidates [5, 51–53]. Antigen-binding fragments (Fabs) of antibodies consist of two amino acid chains, referred to as the *heavy* and *light* chains. Each chain is further divided into four framework (FR) regions and three complementarity-determining regions (CDRs). While FR regions are typically conserved, CDRs exhibit significant variation in their amino acid composition and are generally the primary determinants of binding affinity to the target antigen. When designing novel antibodies for a specific antigen, the typical approach is to explore alternative CDRs that could produce a new, functional antibody with high binding affinity to the target [32, 49, 50, 54].

Recently, several deep learning methods have been developed for targeted antibody design, framing CDR prediction as an *infilling* task [32, 54–59]. These models predict missing CDR regions, represented by *MASK* tokens, using the amino acid sequences of both the antigen and the antibody's FR regions. While prior approaches often rely on structural data, this information is scarce and challenging to obtain [60]. In contrast, we fine-tune MAMMAL for the targeted antibody design task using only the sequence data from the antigen and the antibody's FR regions.

The targeted antibody design task benchmark is based on the SAbDab dataset [60]. Following the data processing outlined in [32], we filtered out samples with missing CDRs to enable direct comparison, even though MAMMAL supports samples that contain missing CDRs. Consistent with [32], we randomly partitioned the dataset into training, validation, and test folds while ensuring that samples with similar heavy CDR3 sub-sequences remained in the same fold. As demonstrated in Tables 2 and S4, MAMMAL shows superior amino acid recovery across all masked CDRs. Notably, in CDRH3, the most variable region, it exhibits a remarkable improvement of 19%.

3.5 Antibody-Antigen Binding Prediction

Accurate prediction of antigen-antibody binding can enhance the design and optimization of therapeutic antibodies, leading to improved efficacy and specificity. We employ the human epidermal growth factor receptor 2 (HER2) dataset [61] as a benchmark for predicting antibody-antigen binding. HER2 is a key target for certain types of breast and stomach cancers. The dataset includes variations of the clinically approved therapeutic antibody trastuzumab and their corresponding affinities for the HER2 antigen. The dataset comprises 8,935 binding and 25,114 non-binding trastuzumab CDR H3 mutants, each with up to 10 mutations, following de-duplication and the removal of samples labeled as both binding and non-binding.

The HER2 dataset was divided into train (70%), validation (15%) and test (15%) sets. Fine-tuning involved three concurrent tasks: mask infilling for the antibody heavy chains, and two classification tasks for antibody-antigen binding prediction: one utilizing the heavy chain sequence and the other based on the CDR3 subsequence. We focus on the heavy chain classification task for performance evaluation. As depicted in table 2, our model achieved an average AUC of 0.879, slightly surpassing the SOTA, which incorporated structural data that our model did not.

3.6 T-Cell Receptor-Epitope Binding

T-cell receptor (TCR) binding to immunogenic peptides (epitopes) presented by major histocompatibility complex (MHC) molecules is a critical mechanism in the adaptive immune system, essential for antigen recognition and triggering immune responses. The T-cell receptor (TCR) repertoire exhibits considerable diversity, consisting of an α-chain and a β-chain that function together to enable T cells to recognize a wide array of epitopes. The β-chain is especially significant, as it is crucial for the early stages of T-cell development and possesses greater variability, which enhances the TCR's capacity to identify diverse pathogens effectively. However, understanding the specific interactions between TCRs and epitopes remains a significant challenge due to the vast variability in TCR sequences. Accurate prediction of TCR-peptide binding from sequence data could revolutionize immunology by offering deeper insights into a patient's immune status and disease history. This capability holds potential applications in personalized immunotherapy, early diagnosis, and the treatment of diseases such as cancer and autoimmune disorders. In silico tools designed to model TCR-peptide interactions could also facilitate the study of therapeutic T-cell efficacy and assess cross-reactivity risks, presenting a transformative opportunity for precision medicine.

We evaluated the model on the task of predicting TCR-epitope binding from sequence data using the Weber benchmark ([34], [62]), which consists of 47,182 TCR β-chain epitope pairs. This dataset covers 192 distinct epitopes and includes 23,139 unique TCR β-chain sequences, with 50% of the pairs serving as negative samples created by pairing TCR sequences with epitopes they are not known to bind. The dataset also includes the CDR3 subsequence for each TCR β-chain , the most hypervariable region of the chain. We used 10-fold cross-validation, using folds from the original TITAN paper [34]. Fine-tuning involved three concurrent tasks: TCR β-chain mask infilling and two classification tasks: (i) TCR β-chain epitope binding prediction and (ii) TCR β-chain -CDR3 epitope binding prediction. Here, we report the performance only for the TCR β-chain epitope binding prediction task. As depicted in table 2, our model achieved an average AUROC of 0.879, representing a statistically significant improvement of 2% over the SOTA, as our result falls outside the SOTA's confidence interval.

3.7 Protein Protein Interaction - ΔΔG Prediction

An important factor in drug design is binding affinity, commonly measured by the equilibrium dissociation constant, K_D, which is related to the Gibbs free energy ΔG through the equation

$$\Delta G = kT \ln(K_D), \tag{1}$$

where k is the Boltzmann constant and T is the temperature [63].

The effect of mutating several residues in a protein complex on binding affinity can be quantified by the difference in ΔG between the mutant and the reference (wild-type) complex. This difference is expressed as

$$\Delta\Delta G = \Delta G_{\text{mutant}} - \Delta G_{\text{wild-type}}.$$

Predicting $\Delta\Delta G$ is a central focus of numerous research efforts [64–66].

The SKEMPI dataset [63] provides information on changes in thermodynamic parameters, including ΔG, and kinetic rate constants due to mutations in protein-protein (PP) complexes whose structures are available in the Protein Data Bank [67]. This dataset is extensively utilized in the literature for predicting the effects of mutations on binding affinity, particularly in the context of $\Delta\Delta G$. A subset of SKEMPI comprising 1,131 samples of single mutations, S1131, is widely used. We adopt S1131 as our benchmark for predicting protein-protein $\Delta\Delta G$ and follow the common practice of reporting 10-fold cross-validation performance on this subset. The input query for our model includes the reference "wild-type" version of the complex and the corresponding mutated version, comprising only sequence data without any structural information. We leverage MAMMAL's support for floating-point scalars to predict continuous $\Delta\Delta G$ in a regression task setting. Performance results are presented in Table 2. As shown in Table 2, our model achieved an average Pearson correlation of 0.852, significantly exceeding the previous sequence-only SOTA of 0.663. Compared to models that incorporate structural data, our model's performance remains competitive, falling just 1.6% short of the SOTA performance of 0.866 [64].

3.8 Drug-Target Interaction

Predicting drug-target binding affinity plays a crucial role in the early stages of drug discovery. Traditionally, binding affinities are measured through high-throughput screening experiments, which, while accurate, are resource-intensive and limited in their scalability to evaluate large sets of drug candidates. In this task, we focus on predicting binding affinities using pKd, the negative logarithm of the dissociation constant, which reflects the strength of the interaction between a small molecule (drug) and a protein (target). We utilize the PEER(Protein sEquence undERstanding) benchmark [36] for drug-target interaction (DTI) prediction. This benchmark leverages data from the BindingDB dataset [68], with a specific test split that holds out four protein classes - estrogen receptor (ER), G-protein-coupled receptors (GPCR), ion channels, and receptor tyrosine kinases - for assessing generalization performance on unseen classes.

For model fine-tuning, we conducted hyperparameter optimization, selecting an initial learning rate of 0.0004, with no dropout and no weight decay. We standardized the pKd values based on the mean and standard deviation of the training set. For evaluation, we transformed the predicted values back to their original scale. As shown in Table 2, our model achieved an average NRMSE of 0.906, demonstrating a solid improvement of 3.8% over the SOTA reported by [36].

4 Discussion

Artificial intelligence (AI) holds great promise for transforming the drug discovery process by enhancing efficiency, accuracy, and speed. Developed with this goal in mind, our proposed method enables the creation of AI models capable of handling diverse tasks across multiple data domains. MAMMAL reformulates tasks as sequence-to-sequence problems and introduces several key architectural enhancements: support for both encoder-only and encoder-decoder modes, a multi-domain extensible syntax for inputs and outputs, and the direct handling of numerical values through continuous token embeddings. MAMMAL has been applied to pretrain model `ibm/biomed .omics.bl.sm.ma-ted-458m` while aligning inputs across diverse datasets - including small molecule, protein, antibody, and gene expression data-using a variety of pretraining tasks. This multi-align approach enables the integration of cross-domain pharmaceutical knowledge into a single model and facilitates effective transfer learning to fine-tuned models for downstream applications. Demonstrated state-of-the-art performance of the multi-align fine-tuned models across diverse tasks, spanning multiple data domains and stages of the drug discovery pipeline, strongly supports the potential of the MAMMAL approach.

Supporting both encoder-only and encoder-decoder modes is motivated by the observation that different tasks benefit from distinct architectures: generative tasks with variable output lengths are well-suited to an encoder-decoder approach, while classification and regression tasks align more effectively with an encoder-only setup. Sharing the encoder stack across both modes optimizes it jointly across tasks, allowing each to benefit from the collective knowledge embedded within the encoder. Additionally, the extensible molecular prompt language enables flexible task formulation across diverse domains, unlike models constrained by fixed prompt structures. This flexibility enhances applicability across various domains, making it a versatile tool for researchers from different fields and promoting interdisciplinary collaboration.

Drug discovery is often hindered by a lack of large, high-quality datasets, particularly in biomedicine, where data generation and collection are costly and face challenges related to privacy, lack of standardization, and ethical constraints. A planned extension of our approach involves the support of free-text input, enabling pretraining on extensive biomedical text sources like PubMed as well as the incorporation of free-text segments into prompts of fine-tuned models. Free text provides rich, nuanced context and allows users greater flexibility beyond predefined formats. The integration of large language models, which capture human narratives, with our newly introduced multi-aligned model-demonstrated to effectively encode complex interactions within protein complexes, between proteins and small molecules, and between small molecules and cells, including their gene expression levels-holds promise of creating a model capable of uncovering new scientific knowledge.

We are pleased to announce the open-source release of the the code and the pretrained model weights for `ibm/biomed.omics.bl.sm.ma-ted-458m`, inviting the community to explore, apply, and contribute to its ongoing development. By making MAMMAL accessible, we aim to foster collaboration that enriches the platform through the addition of new pretraining tasks and the integration of new domains, such as DNA sequences. We believe that such collaborative efforts will strengthen and enhance the capabilities of MAMMAL, broadening its applicability in the field of biomedicine and ultimately leading to new discoveries in drug research.

5 Data Availability

All datasets used in this study are publicly available.
Cell Type. Dataset (Zheng68k) was obtained from https://www.10xgenomics.com/datasets/fresh-68-k-pbm-cs-donor-a-1-standard-1-1-0 (file: https://cf.10xgenomics.com/samples/cell-exp/1.1.0/fresh_68k_pbmc_donor_a/fresh_68k_pbmc_donor_a_filtered_gene_bc_matrices.tar.gz)

BBBP and ClinTox benchmarks were obtained from https://github.com/IBM/molformer/tree/main/data that points to https://ibm.ent.box.com/v/MoLFormer-data (file: `finetune_datasets.zip`).

Cancer-Drug Response 1 and 2. GDSC1 and GDSC2 benchmarks were accessed with random splits from the TDC library (https://pypi.org/project/PyTDC/). **Cancer-Drug Response 3.** Benchmark obtained from DeepCDR [31] git repository (https://github.com/kimmo1019/DeepCDR/tree/master/data).

DTI. Benchmark was published by [36] and available in https://torchdrug.ai/docs/api/datasets.html#bindingdb

Ab Infilling. Data is taken from [32], which provides a preprocessed subset of the publicly available SAbDab database [60]. The preprocessing pipeline includes a similarity-based clustering for the data splits and sample level filtering which excludes samples that are considered in valid in [32]. For additional information, we refer to [32] and the publicly available codebase, https://github.com/THUNLP-MT/dyMEAN.

PPI $\Delta\Delta G$. The SKEMPI S1131 dataset of non-redundant single mutations was derived from SKEMPI [63] in [69] and can be downloaded from https://zhanggroup.org/BindProfX/download/.

PPI. The Weber TCR binding dataset was downloaded from [62], and the HER2 antibody-antigen binding dataset was taken from the original paper [61] github repository [70].

Pretraining. `ibm/biomed.omics.bl.sm.ma-ted-458m` was pre-trained over OAS [23], Uniref90 [71], Zinc [24], PubChem [25] and CELLxGENE [26]. Appendix C describes the pre-processing steps applied.

6 Code Availability

The model architecture, fine-tuning framework, and end-to-end examples are publicly available at https://github.com/BiomedSciAI/biomed-multi-alignment. This repository provides comprehensive resources for utilizing the model, including instructions for fine-tuning, and performing inference on various tasks. The pretrained model weights and tokenizer can be accessed via the Hugging Face model hub at https://huggingface.co/ibm/biomed.omics.bl.sm.ma-ted-458m.

References

[1] Paul, S.M., Mytelka, D.S., Dunwiddie, C.T., Persinger, C.C., Munos, B.H., Lindborg, S.R., Schacht, A.L.: How to improve r&d productivity: the pharmaceutical industry's grand challenge. Nature reviews Drug discovery **9**(3), 203–214 (2010)

[2] DiMasi, J.A., Grabowski, H.G., Hansen, R.W.: Innovation in the pharmaceutical industry: new estimates of r&d costs. Journal of health economics **47**, 20–33 (2016)

[3] Wouters, O.J., McKee, M., Luyten, J.: Estimated research and development investment needed to bring a new medicine to market, 2009-2018. Jama **323**(9), 844–853 (2020)

[4] Southey, M.W., Brunavs, M.: Introduction to small molecule drug discovery and preclinical development. Frontiers in Drug Discovery **3**, 1314077 (2023)

[5] Lu, R.-M., Hwang, Y.-C., Liu, I.-J., Lee, C.-C., Tsai, H.-Z., Li, H.-J., Wu, H.-C.: Development of therapeutic antibodies for the treatment of diseases. Journal of biomedical science **27**, 1–30 (2020)

[6] Sadybekov, A.V., Katritch, V.: Computational approaches streamlining drug discovery. Nature **616**(7958), 673–685 (2023)

[7] Huang, D., Yang, M., Wen, X., Xia, S., Yuan, B.: Ai-driven drug discovery:: Accelerating the development of novel therapeutics in biopharmaceuticals. Journal of Knowledge Learning and Science Technology ISSN: 2959-6386 (online) **3**(3), 206–224 (2024)

[8] Son, A., Park, J., Kim, W., Yoon, Y., Lee, S., Park, Y., Kim, H.: Revolutionizing molecular design for innovative therapeutic applications through artificial intelligence. Molecules **29**(19), 4626 (2024)

[9] Baslan, T., Hicks, J.: Unravelling biology and shifting paradigms in cancer with single-cell sequencing. Nature Reviews Cancer **17**(9), 557–569 (2017)

[10] Ofengeim, D., Giagtzoglou, N., Huh, D., Zou, C., Yuan, J.: Single-cell rna sequencing: unraveling the brain one cell at a time. Trends in molecular medicine **23**(6), 563–576 (2017)

[11] Rozenblatt-Rosen, O., Stubbington, M.J., Regev, A., Teichmann, S.A.: The human cell atlas: from vision to reality. Nature **550**(7677), 451–453 (2017)

[12] Potter, S.S.: Single-cell rna sequencing for the study of development, physiology and disease. Nature Reviews Nephrology **14**(8), 479–492 (2018)

[13] Sande, B., Lee, J.S., Mutasa-Gottgens, E., Naughton, B., Bacon, W., Manning, J., Wang, Y., Pollard, J., Mendez, M., Hill, J., *et al.*: Applications of single-cell rna sequencing in drug discovery and development. Nature Reviews Drug Discovery **22**(6), 496–520 (2023)

[14] Dann, E., Teeple, E., Elmentaite, R., Meyer, K.B., Gaglia, G., Nestle, F., Savova, V., Rinaldis, E., Teichmann, S.A.: Estimating the

impact of single-cell rna sequencing of human tissues on drug target validation. medRxiv (2024) https://doi.org/10.1101/2024.04.04.24305313 https://www.medrxiv.org/content/early/2024/10/22/2024.04.04.24305313.full.pdf

[15] Tang, X., Dai, H., Knight, E., Wu, F., Li, Y., Li, T., Gerstein, M.: A survey of generative ai for de novo drug design: new frontiers in molecule and protein generation. Briefings in Bioinformatics **25**(4), 338 (2024)

[16] Shanehsazzadeh, A., Bachas, S., McPartlon, M., Kasun, G., Sutton, J.M., Steiger, A.K., Shuai, R., Kohnert, C., Rakocevic, G., Gutierrez, J.M., et al.: Unlocking de novo antibody design with generative artificial intelligence. bioRxiv, 2023–01 (2023)

[17] Swanson, K., Liu, G., Catacutan, D.B., Arnold, A., Zou, J., Stokes, J.M.: Generative ai for designing and validating easily synthesizable and structurally novel antibiotics. Nature Machine Intelligence **6**(3), 338–353 (2024)

[18] Athaya, T., Ripan, R.C., Li, X., Hu, H.: Multimodal deep learning approaches for single-cell multi-omics data integration. Briefings in Bioinformatics **24**(5), 313 (2023)

[19] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems (2017)

[20] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020)

[21] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (2019). https://doi.org/10.18653/v1/N19-1423 . https://aclanthology.org/N19-1423

[22] Consortium, T.U.: UniProt: the Universal Protein Knowledgebase in 2023. Nucleic Acids Research **51**(D1), 523–531 (2022) https://doi.org/10.1093/nar/gkac1052 https://academic.oup.com/nar/article-pdf/51/D1/D523/48441158/gkac1052.pdf

[23] Olsen, T.H., Boyles, F., Deane, C.M.: Observed antibody space: A diverse database of cleaned, annotated, and translated unpaired and paired antibody sequences. Protein Science **31**(1), 141–146 (2022) https://doi.org/10.1002/pro.4205 https://onlinelibrary.wiley.com/doi/pdf/10.1002/pro.4205

[24] Tingle, B.I., Tang, K.G., Castanon, M., Gutierrez, J.J., Khurelbaatar, M., Dandarchuluun, C., Moroz, Y.S., Irwin, J.J.: Zinc-22-a free multi-billion-scale database of tangible compounds for ligand discovery. Journal of Chemical Information and Modeling **63**(4), 1166–1176 (2023) https://doi.org/10.1021/acs.jcim.2c01253 https://doi.org/10.1021/acs.jcim.2c01253. PMID: 36790087

[25] Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B.A., Thiessen, P.A., Yu, B., Zaslavsky, L., Zhang, J., Bolton, E.E.: PubChem 2023 update. Nucleic Acids Research **51**(D1), 1373–1380 (2022) https://doi.org/10.1093/nar/gkac956 https://academic.oup.com/nar/article-pdf/51/D1/D1373/48441598/gkac956.pdf

[26] Biology, C.S.-C., Abdulla, S., Aevermann, B., Assis, P., Badajoz, S., Bell, S.M., Bezzi, E., Cakir, B., Chaffer, J., Chambers, S., et al.: Cz cellxgene discover: A single-cell data platform for scalable exploration, analysis and modeling of aggregated data. BioRxiv, 2023–10 (2023)

[27] Szklarczyk, D., Kirsch, R., Koutrouli, M., Nastou, K., Mehryary, F., Hachilif, R., Gable, A.L., Fang, T., Doncheva, N., Pyysalo, S., Bork, P., Jensen, L., von Mering, C.: The STRING database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. Nucleic Acids Research **51**(D1), 638–646 (2022) https://doi.org/10.1093/nar/gkac1000 https://academic.oup.com/nar/article-pdf/51/D1/D638/48440966/gkac1000.pdf

[28] Xu, J., Zhang, A., Liu, F., Chen, L., Zhang, X.: Ciform as a transformer-based model for cell-type annotation of large-scale single-cell rna-seq data. Briefings in Bioinformatics **24**(4), 195 (2023)

[29] Ross, J., Belgodere, B., Chenthamarakshan, V., Padhi, I., Mroueh, Y., Das, P.: Large-scale chemical language representations capture molecular structure and properties. Nature Machine Intelligence **4**(12), 1256–1264 (2022)

[30] Chaves, J.M.Z., Wang, E., Tu, T., Vaishnav, E.D., Lee, B., Mahdavi, S.S., Semturs, C., Fleet, D., Natarajan, V., Azizi, S.: Tx-llm: A large language model for therapeutics. arXiv preprint arXiv:2406.06316 (2024)

[31] Liu, Q., Hu, Z., Jiang, R., Zhou, M.: Deepcdr: a hybrid graph convolutional network for predicting cancer drug response. Bioinformatics **36**(Supplement_2), 911–918 (2020)

[32] Kong, X., Huang, W., Liu, Y.: End-to-end full-atom antibody design. arXiv preprint arXiv:2302.00203 (2023)

[33] Jing, H., Gao, Z., Xu, S., Shen, T., Peng, Z., He, S., You, T., Ye, S., Lin, W., Sun, S.: Accurate prediction of antibody function and structure using bio-inspired antibody language model. Briefings in Bioinformatics **25**(4), 245 (2024)

[34] Weber, A., Born, J., Rodriguez Martínez, M.: Titan: T-cell receptor specificity prediction with bimodal attention networks. Bioinformatics **37**(Supplement_1), 237–244 (2021)

[35] Jin, R., Ye, Q., Wang, J., Cao, Z., Jiang, D., Wang, T., Kang, Y., Xu, W., Hsieh, C.-Y., Hou, T.: Attabseq: an attention-based deep learning prediction method for antigen–antibody binding affinity changes based on protein sequences. Briefings in Bioinformatics **25**(4), 304 (2024)

[36] Xu, M., Zhang, Z., Lu, J., Zhu, Z., Zhang, Y., Chang, M., Liu, R., Tang, J.: Peer: a comprehensive and multi-task benchmark for protein sequence understanding. Advances in Neural Information Processing Systems **35**, 35156–35173 (2022)

[37] Qi, R., Ma, A., Ma, Q., Zou, Q.: Clustering and classification methods for single-cell rna-sequencing data. Briefings in bioinformatics **21**(4), 1196–1208 (2020)

[38] Cui, H., Wang, C., Maan, H., Pang, K., Luo, F., Duan, N., Wang, B.: scgpt: toward building a foundation model for single-cell multi-omics using generative ai. Nature Methods, 1–11 (2024)

[39] Yang, F., Wang, W., Wang, F., Fang, Y., Tang, D., Huang, J., Lu, H., Yao, J.: scbert as a large-scale pretrained deep language model for cell type annotation of single-cell rna-seq data. Nature Machine Intelligence **4**(10), 852–866 (2022)

[40] Zheng, G.X., Terry, J.M., Belgrader, P., Ryvkin, P., Bent, Z.W., Wilson, R., Ziraldo, S.B., Wheeler, T.D., McDermott, G.P., Zhu, J., *et al.*: Massively parallel digital transcriptional profiling of single cells. Nature communications **8**(1), 14049 (2017)

[41] Theodoris, C.V., Xiao, L., Chopra, A., Chaffin, M.D., Al Sayed, Z.R., Hill, M.C., Mantineo, H., Brydon, E.M., Zeng, Z., Liu, X.S., Ellinor, P.T.: Transfer learning enables predictions in network biology. Nature **618**(7965), 616–624 (2023) https://doi.org/10.1038/s41586-023-06139-9

[42] Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V.: Moleculenet: a benchmark for molecular machine learning. Chemical science **9**(2), 513–530 (2018)

[43] Barretina, J., Caponigro, G., Stransky, N., Venkatesan, K., Margolin, A.A., Kim, S., Wilson, C.J., Lehár, J., Kryukov, G.V., Sonkin, D., *et al.*: The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity. Nature **483**(7391), 603–607 (2012)

[44] Yang, W., Soares, J., Greninger, P., Edelman, E.J., Lightfoot, H., Forbes, S., Bindal, N., Beare, D., Smith, J.A., Thompson, I.R., *et al.*: Genomics of drug sensitivity in cancer (gdsc): a resource for therapeutic biomarker discovery in cancer cells. Nucleic acids research **41**(D1), 955–961 (2012)

[45] Lind, A.P., Anderson, P.C.: Predicting drug activity against cancer cells by random forest models based on minimal genomic information and chemical properties. PloS one **14**(7), 0219774 (2019)

[46] Liu, X., Song, C., Huang, F., Fu, H., Xiao, W., Zhang, W.: Graphcdr: a graph neural network method with contrastive learning for cancer drug response prediction. Briefings in Bioinformatics **23**(1), 457 (2022)

[47] Huang, K., Fu, T., Gao, W., Zhao, Y., Roohani, Y., Leskovec, J., Coley, C.W., Xiao, C., Sun, J., Zitnik, M.: Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. arXiv preprint arXiv:2102.09548 (2021)

[48] Weinstein, J.N., Collisson, E.A., Mills, G.B., Shaw, K.R., Ozenberger, B.A., Ellrott, K., Shmulevich, I., Sander, C., Stuart, J.M.: The cancer genome atlas pan-cancer analysis project. Nature genetics **45**(10), 1113–1120 (2013)

[49] Hummer, A.M., Abanades, B., Deane, C.M.: Advances in computational structure-based antibody design. Current opinion in structural biology **74**, 102379 (2022)

[50] Chiu, M., Goulet, D., Teplyakov, A., Gilliland, G.: Antibody structure and function: the basis for engineering therapeutics. Antibodies (Basel) 8 (4) (2019)

[51] Basu, K., Green, E.M., Cheng, Y., Craik, C.S.: Why recombinant antibodies - benefits and applications. Current opinion in biotechnology **60**, 153–158 (2019)

[52] Carter, P.J., Lazar, G.A.: Next generation antibody drugs: pursuit of the'high-hanging fruit'. Nature Reviews Drug Discovery **17**(3), 197–223 (2018)

[53] Beck, A., Goetsch, L., Dumontet, C., Corvaïa, N.: Strategies and challenges for the next generation of antibody–drug conjugates. Nature reviews Drug discovery **16**(5), 315–337 (2017)

[54] Saka, K., Kakuzaki, T., Metsugi, S., Kashiwagi, D., Yoshida, K., Wada, M., Tsunoda, H., Teramoto, R.: Antibody design using lstm based deep generative model from phage display library for affinity maturation. Scientific reports **11**(1), 5852 (2021)

[55] Jin, W., Wohlwend, J., Barzilay, R., Jaakkola, T.: Iterative refinement graph neural network for antibody sequence-structure co-design. arXiv preprint arXiv:2110.04624 (2021)

[56] Jin, W., Barzilay, R., Jaakkola, T.: Antibody-antigen docking and design via hierarchical equivariant refinement. arXiv preprint arXiv:2207.06616 (2022)

[57] Luo, S., Su, Y., Peng, X., Wang, S., Peng, J., Ma, J.: Antigen-specific antibody design and optimization with diffusion-based generative models for protein structures. Advances in Neural Information Processing Systems **35**, 9754–9767 (2022)

[58] Kong, X., Huang, W., Liu, Y.: Conditional antibody design as 3d equivariant graph translation. arXiv preprint arXiv:2208.06073 (2022)

[59] Zhou, X., Xue, D., Chen, R., Zheng, Z., Wang, L., Gu, Q.: Antigen-specific antibody design via direct energy-based preference optimization. arXiv preprint arXiv:2403.16576 (2024)

[60] Dunbar, J., Krawczyk, K., Leem, J., Baker, T., Fuchs, A., Georges, G., Shi, J., Deane, C.M.: Sabdab: the structural antibody database. Nucleic acids research **42**(D1), 1140–1146 (2014)

[61] Mason, D.M., Friedensohn, S., Weber, C.R., Jordi, C., Wagner, B., Meng, S.M., Ehling, R.A., Bonati, L., Dahinden, J., Gainza, P., Correia, B.E., Reddy, S.T.: Optimization of therapeutic antibodies by predicting antigen specificity from antibody sequence via deep learning. Nature Biomedical Engineering **5**(6), 600–612 (2021) https://doi.org/10.1038/s41551-021-00699-9 . 169 citations (Semantic Scholar/DOI) [2024-10-02] Number: 6 Publisher: Nature Publishing Group. Accessed 2024-10-02

[62] TCR-Epitope Binding. https://tdcommons.ai/multi_pred_tasks/tcrepitope Accessed 2024-10-02

[63] Jankauskaitė, J., Jiménez-García, B., Dapkūnas, J., Fernández-Recio, J., Moal, I.H.: Skempi 2.0: an updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation. Bioinformatics **35**(3), 462–469 (2019)

[64] Liu, X., Feng, H., Lü, Z., Xia, K.: Persistent tor-algebra for protein–protein interaction analysis. Briefings in Bioinformatics **24**(2), 046 (2023)

[65] Wang, M., Cang, Z., Wei, G.-W.: A topology-based network tree for the prediction of protein–protein binding affinity changes following mutation. Nature Machine Intelligence **2**(2), 116–123 (2020)

[66] Guo, Z., Yamaguchi, R.: Machine learning methods for protein-protein binding affinity prediction in protein design. Frontiers in Bioinformatics **2**, 1065703 (2022)

[67] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The Protein Data Bank. Nucleic Acids Research **28**(1), 235–242 (2000) https://doi.org/10.1093/nar/28.1.235 https://academic.oup.com/nar/article-pdf/28/1/235/9895144/280235.pdf

[68] Gilson, M.K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., Chong, J.: BindingDB in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res **44**(D1), 1045–53 (2015)

[69] Xiong, P., Zhang, C., Zheng, W., Zhang, Y.: Bindprofx: assessing mutation-induced binding affinity change by protein interface profiles with pseudo-counts. Journal of molecular biology **429**(3), 426–434 (2017)

[70] Jan: Dahjan/DMS_opt. original-date: 2020-06-24T09:29:12Z. https://github.com/dahjan/DMS_opt Accessed 2024-10-27

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

[72] Golts, A., Raboh, M., Shoshan, Y., Polaczek, S., Rabinovici-Cohen, S., Hexter, E.: Fusemedml: a framework for accelerated discovery in machine learning based biomedicine. Journal of Open Source Software **8**(81), 4943 (2023) https://doi.org/10.21105/joss.04943

Appendix A Architecture - additional details

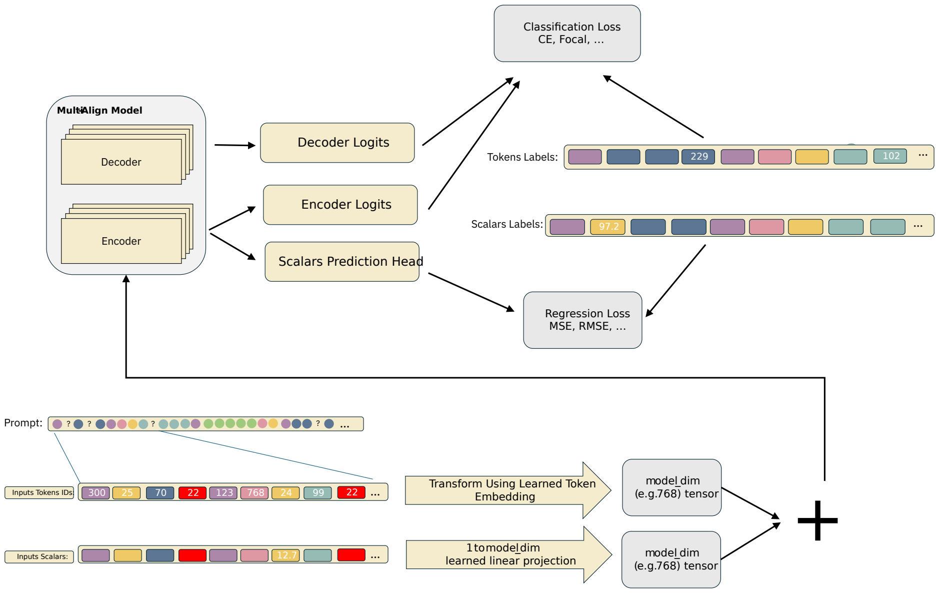


Fig. A1 A prompt, consisting of both token ids and scalars is processed and enters the encoder. Both the encoder and the decoder output logits which are used for classification loss. Additionally, the output of the encoder is sent to a (learned) scalars prediction head which allows to predict scalars for any subset of the tokens, and is used in the regression loss. In this illustration, a single scalar input (”12.7”) is being used, and a single scalars outputs are predicted by the model (”97.2”). However, the method fully supports an arbitrary number of input scalars and outputs.

One of the key aspects in MAMMAL method is the built in support for scalars inputs and outputs. Figure A1 illustrates how this is achieved.

A user prompt, usually expressed as a single text line, is processed into 2 input sequences: a. Input token IDs, which is a sequence of integer values representing tokens in the vocabulary b. a sequence of inputs scalars (by convention, containing NaNs for positions for which no input scalar is provided).

The input tokens ids are transformed using a learned token embedding, and the input scalars are transformed using a learned linear transformation which projects each single scalar element into the model dimension (e.g. 768).

Both representations are added (not concatenated) and fed into the encoder stack. Using this approach, both tasks that use encoder-only mode and encoder-decoder mode benefit from the ability to get as input an arbitrary number of scalars (at most as many as the number of tokens that are being fed in).

For scalars outputs (gene expression, binding free enery, etc.) the encoder stack has an additional prediction head, which outputs a scalar value for every input element. How to deal with scalars outputs in locations that there is no scalar label is up to the user choice, but the default is to ignore those.

The support of scalars outputs in the encoder-decoder mode is an improvement that we intend to add in future generations of the model/method.

Appendix B Prompt Syntax

A typical prompt is built as a combination of entities of the following types:

- **SubSequence** : A sequence of amino acids or other chemical representations, such as SMILES, which may encompass a full sequence or a specific region. A SubSequence can begin with two special tokens: ⟨SUBMOLECULAR_ENTITY⟩ , followed by a token indicating the SubSequence type (e.g., ⟨CDR3_REGION⟩). These tokens are optional and may be omitted when only one SubSequence is present.
- Molecule: A complete molecule, such as a protein chain or small molecule, which may contain multiple SubSequences corresponding to sub-regions within the molecule. Each Molecule is initiated with two special tokens - a general token indicating the entity's hierarchical level, ⟨MOLECULAR_ENTITY_TOKEN⟩, followed by a token specifying the molecule type (e.g., ⟨MOLECULAR_ENTITY_EPITOPE⟩). Additionally, Molecules can be marked with natural start and end tokens to denote instances where truncation has occurred, either in the original database or due to sequence length constraints.
- MolecularSystem: A quaternary structure consisting of multiple Molecules, denoted by the ⟨COMPLEX_ENTITY⟩ special token.
- GlobalSystem: A system comprising multiple interacting MolecularSystem entities.
- Attribute: A representation of properties or interactions among the entities.

In predictive tasks, relevant attribute values are masked, while in generative or masked language modeling (MLM) tasks, spans within SubSequences or entire SubSequences are masked. Tokens denoting entity types can also be masked for type prediction tasks. Additionally, each entity may possess alternative expressions or mutations, which can be employed for comparison tasks.

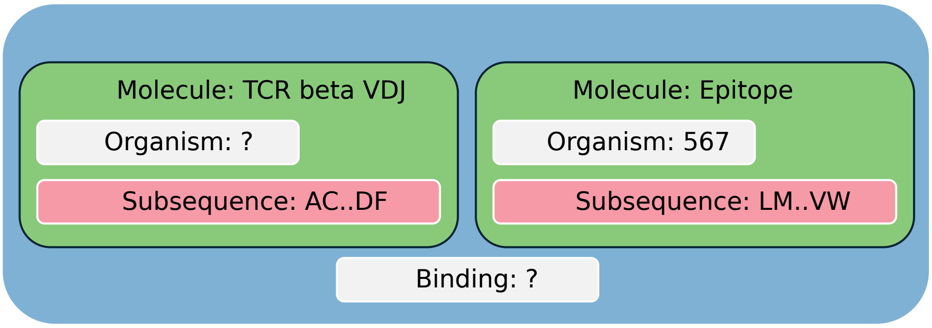


Fig. B2 Entity hierarchy for the task of binding prediction of two proteins, and organism prediction of the first one.

Example 1: Illustrated in Figure B2. Given two interacting molecules – variable region of a TCR beta chain of an unspecified organism and an epitope of organism 567, find whether they bind, and to which organism the first molecule belongs.:

- encoder inputs = ⟨@TOKENIZER-TYPE=AA⟩ ⟨BINDING_AFFINITY_CLASS⟩ ⟨SENTINEL_ID_0⟩ ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TCR_BETA_VDJ⟩ ⟨ATTRIBUTE_ORGANISM⟩⟨SENTINEL_ID_1⟩ AC...DF ⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY_EPITOPE⟩⟨ATTRIBUTE_ORGANISM⟩⟨5⟩⟨6⟩⟨7⟩⟨8⟩ LM...VW ⟨EOS⟩
- labels = ⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩⟨1⟩⟨SENTINEL_ID_1⟩⟨2⟩⟨3⟩⟨4⟩ ⟨EOS⟩

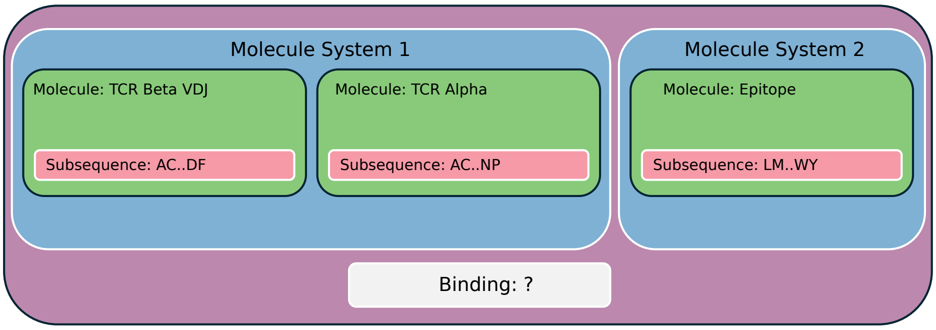


Fig. B3 Entity hierarchy for the task of binding prediction of a TCR and an epitope. ”Molecule System 1” represents the TCR complex, ”Molecule System 2” represents the antigen, and the entire prompt represents their interaction.

Example 2 : Illustrated in Figure B3. Given a complex entity, T-cell receptor consisting of alpha and beta chains, and an epitope, predict if they bind.:

- encoder inputs = ⟨@TOKENIZER-TYPE=AA⟩⟨BINDING_AFFINITY_CLASS⟩ ⟨SENTINEL_ID_0⟩ ⟨COMPLEX_ENTITY⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TCR_BETA_VDJ⟩ AB...DF ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TCR_ALPHA⟩ AC...NP ⟨COMPLEX_ENTITY⟩ ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_EPITOPE⟩LM...WY ⟨EOS⟩
- labels = ⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩

Example: Given two binding chains – TCR beta chain and an epitope, unmask 3 spans within the beta chain:

- labels = ⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩AC⟨SENTINEL_ID_1⟩ AD⟨SENTINEL_ID_2⟩CD⟨EOS⟩
- encoder inputs = ⟨@TOKENIZER-TYPE=AA⟩ ⟨BINDING_AFFINITY_CLASS⟩⟨1⟩ ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_EPITOPE⟩⟨SEQUENCE_NATURAL_START⟩ LM...WY⟨SEQUENCE_NATURAL_END⟩⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY_TCR_BETA_VDJ⟩ ⟨SENTINEL_ID_0⟩**C...D**⟨SENTINEL_ID_1⟩ **DF**⟨SENTINEL_ID_2⟩**FDF** ⟨EOS⟩

Example 3: (for multitokenizer): DTI

- labels = ⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩
- encoder inputs = ⟨@TOKENIZER-TYPE=AA⟩⟨BINDING_AFFINITY_CLASS⟩ ⟨SENTINEL_ID_0⟩⟨@TOKENIZER-TYPE=SMILES@MAX-LEN=10⟩⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY_SMALL_MOLECULE⟩**CCC=CCC** ⟨@TOKENIZER-TYPE=AA@MAXLEN=15⟩⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY_EPITOPE⟩ **LMNPQRSTUVWY** ⟨EOS⟩

Examples showing how a prompt is created for several downstream tasks can be found in Table S1

B.1 Modular Tokenizer and Meta Tokens

To support multiple modalities within a single prompt we have developed ”Modular Tokenizer” which allows to utilize different tokenizers within a single prompt by mapping tokens from different domains (like SMILES carbon ”C” and amino acid cysteine ”C”) to the same ID space.

We use ”Meta Tokens” of the format ⟨@TOKENIZER-TYPE=...⟩ to indicate that everything following this meta token, up to the next meta token (or the end of the prompt) should be tokenized with the defined tokenizer. For example, ⟨@TOKENIZER-TYPE=AA⟩ tokenizes amino-acids, while ⟨@TOKENIZER-TYPE=SMILES⟩ can tokenize SMILES. Since all of those ”sub tokenizers” must exist in a single vocabulary space, our modular tokenizer orchestrates that, and provides mechanism to avoid conflicts and for sub-tokenizers to co-exist. Additionally, we support additional instructions within a meta token beyond only expression which (sub) tokenizer should be used. For example, ⟨@TOKENIZER-TYPE=AA@MAX-LEN=1000⟩ allows to restrict the maximum length of the tokenized sequence, which provides more granular control compared to only controlling the overall total max sequence tokenized length. It is worth emphasizing - meta tokens, by themselves, do not get tokenized into any token. They serve as instructions for the modular tokenizer. Further details on the implementation can be found on https://github.com/BiomedSciAI/fuse-med-ml/tree/master/fuse/data/tokenizers/modular_tokenizer

Appendix C Pretraining Details

C.1 Infrastructure

`ibm/biomed.omics.bl.sm.ma-ted-458m` model was trained on an OpenShift cluster. It was trained for three months over two nodes with 16 A100-80G GPUs. The training framework was implemented using FuseMedML [72] and PyTorch, with distributed processing supported by PyTorch Fully Sharded Data Parallel (FSDP) for efficient parallelism.

C.2 Hyperparameters

We train ibm/biomed.omics.bl.sm.ma-ted-458m using AdamW optimizer, with the following hyperparameters: $\beta 1 = 0.9$, $\beta 2 = 0.999$. We use a weight decay of 0.01 and a gradient clipping norm of 1.0. We employ 2K warmup steps until reaching the maximum learning rate and utilize a cosine decay scheduler to decay LR to 10% of the maximum learning rate by the end of training. The maximum sequence length was set per task to be effective yet efficient. When required, instead of naively truncating the end of a sequence, we first wrapped the sequence with special start and end tokens to provide a hint for the model as to whether the beginning or end of a sequence was truncated. Then we randomly cut a random sub-sequence with the required length. Batch sizes were tuned per task given the maximum sequence length to maximize GPU memory utilization.

C.3 Datasets

ibm/biomed.omics.bl.sm.ma-ted-458m was pre-trained using six diverse datasets spanning multiple domains.

UniRef90 [71], one of the clustered databases in UniProt [22] (UniProt Reference Clusters), groups protein sequences that share at least 90% identity and 80% sequence overlap with the longest sequence in each cluster (the seed sequence). This clustering approach reduces redundancy while preserving the diversity of functional sequences, providing a rich protein dataset.

OAS [23] (Observed Antibody Space) offers unpaired antibody sequences, specifically focusing on the variable regions of light or heavy chains. After filtering for sequences with complete variable domains and retaining only the sequences with standard amino acids, we finalized a dataset of approximately 650 million sequences. Each sequence is annotated with its chain type (heavy or light) and species information.

STRING [27] is a database that integrates data from experimental findings, computational predictions, and text mining to describe protein-protein interactions. We curated a dataset of 390 million positive protein interaction pairs, considering only pairs that had a STRING confidence score above 500. Additionally, we curated 390 million pseudo-negative pairs by randomly matching proteins from the same species, resulting in a second dataset of 780 million samples.

CELLxGENE [26], a platform for single-cell transcriptomics data, was used to assemble a dataset of gene expression sequences from individual cells. After filtering for samples labeled as “cell” in the ’suspension_type’ field, we curated a dataset of 30 million samples. These samples were processed and converted into the Genformer format for use in model training.

For small-molecule data, we utilized two main sources: (1) **PubChem** [25], a comprehensive chemical database maintained by NCBI, and (2) **ZINC22** [24], a large library of drug-like molecules. From PubChem, we curated a subset of 80 million “drug-like” molecules, removing duplicates and following Lipinski’s rule of five to ensure druglikeness. Additionally, we sampled 120 million molecules from the ZINC22 database, focusing on small molecules with fewer than 30 heavy atoms to ensure dense coverage of “drug-like” chemical space. This led to a final dataset of 200 million small-molecule examples.

C.4 Multitask Pretraining Approach

`ibm/biomed.omics.bl.sm.ma-ted-458m` was pre-trained for seven tasks simultaneously using a multitask learning approach. Gradients from each task were aggregated before applying optimizer updates. This approach, combined with a custom query system, enables the model to learn from different tasks and domains during co-training.

C.5 Pretraining Tasks

Language Model Tasks. Four language model tasks were defined: (1) amino-acid sequence representation of antibodies based on OAS database [23] (2) amino-acid sequence representation of general proteins based on Uniref90 [22, 71] (3) smiles representation of small molecules based on a mixture of PubChem [25] and Zinc databases [24] (4) Genformer format representations [41] of cell genes based on CELLxGENE [26]. In all language modeling tasks, we employed span-denoising (similar to T5 [20]) with a mean noise span length of 5 and a noise density of 0.15. Additionally, a special token was introduced per entity type to make the model aware of it (e.g., ⟨`MOLECULAR_ENTITY_TYPE_ANTIBODY_HEAVY_CHAIN`⟩). When sequences were available from different species, an additional special token was also introduced (e.g., ⟨`ATTRIBUTE_ORGANISM_HUMAN`⟩)

Antibody Denoise. This task focuses on recovering corrupted antibodies, represented by amino acid sequences. The corrupted sequence is generated by first sampling a value t from the range [1, 500], and then uniformly corrupting the amino acid tokens with a probability proportional to t. The antibody sequences used in this task are sourced from the OAS (Observatory of Antibody Space) dataset.

Protein-Protein Interaction. As part of the pretraining process, two tasks were defined for learning protein-protein interactions: a classification task and a generation task, both utilizing data from the STRING database [27]. Interactions with a score higher than 500 are labeled as positive, while random pairs of proteins are treated as negative interactions. For the classification task, a balanced dataset comprising both positive and negative pairs is used. In the generation task, the model is trained on positive pairs only, where it learns to generate an interacting protein given an input protein.

Appendix D Additional Results

Table S1: Examples of Encoder Inputs and Decoder Outputs for Benchmarks

Bench.	Encoder input	Decoder label
Cell type	⟨@TOKENIZER-TYPE=GENE⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_CELL_GENE_EXPRESSION_RANKED⟩ [MALAT1] [RPL10] . . . [ZNF136] [ZNF514] ⟨CELL_TYPE_CLASS⟩⟨SENTINEL_ID_0⟩⟨EOS⟩	⟨@TOKENIZER-TYPE=CELL_ATTRIBUTES⟩⟨SENTINEL_ID_0⟩[CL:0001062] ⟨EOS⟩
BBBP	⟨@TOKENIZER-TYPE=SMILES⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_SMALL_MOLECULE⟩⟨BBBP⟩⟨SENTINEL_ID_0⟩ ⟨@TOKENIZER-TYPE=SMILES@MAX-LEN=2100⟩ C(Cl)Cl ⟨EOS⟩	⟨@TOKENIZER-TYPE=SMILES⟩ ⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩
ClinTox Toxic	⟨@TOKENIZER-TYPE=SMILES⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_SMALL_MOLECULE⟩⟨TOXICITY⟩⟨SENTINEL_ID_0⟩ ⟨@TOKENIZER-TYPE=SMILES@MAX-LEN=2100⟩ C#CC1(CCCCC1)OC(=O)N ⟨EOS⟩	⟨@TOKENIZER-TYPE=SMILES⟩ ⟨SENTINEL_ID_0⟩⟨0⟩⟨EOS⟩
ClinTox FDA Approval	⟨@TOKENIZER-TYPE=SMILES⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_SMALL_MOLECULE⟩⟨FDA_APPR⟩⟨SENTINEL_ID_0⟩ ⟨@TOKENIZER-TYPE=SMILES@MAX-LEN=2100⟩ C#CC1(CCCCC1)OC(=O)N ⟨EOS⟩	⟨@TOKENIZER-TYPE=SMILES⟩ ⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩
Cancer-Drug Response	⟨@TOKENIZER-TYPE=SCALARS_LITERALS⟩ <MASK > ⟨@TOKENIZER-TYPE=SMILES⟩ ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_SMALL_MOLECULE⟩⟨SMILES_SEQUENCE⟩ CN(C)CCOc. . . [nH]2)cc1 ⟨@TOKENIZER-TYPE=GENES⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_CELL_GENE_EXPRESSION_RANKED⟩ [B2M] [RPL10] . . . [ZBTB16] [ZNF429] ⟨EOS⟩	⟨@TOKENIZER-TYPE=SCALARS_LITERALS⟩3.966226 ⟨@TOKENIZER-TYPE=SMILES⟩ . . . [ZBTB16] [ZNF429] ⟨EOS⟩

Continued on next page

Table S1 – continued from previous page

Bench.	Encoder input	Decoder label
Antibody design	⟨@TOKENIZER-TYPE=AA⟩⟨COMPLEX_ENTITY⟩⟨ATTRIBUTE_ORGANISM⟩⟨ATTRIBUTE_ORGANISM_HUMAN⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TYPE_ANTIBODY_LIGHT_CHAIN⟩AB...CD⟨SENTINEL_ID_0⟩GF...KL⟨SENTINEL_ID_1⟩...TC ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TYPE_ANTIBODY_HEAVY_CHAIN⟩AA...DD⟨SENTINEL_ID_2⟩FK...KF⟨SENTINEL_ID_3⟩...JJ ⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_EPITOPE⟩AB...GK ⟨EOS⟩	⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩ABC⟨SENTINEL_ID_1⟩DDDDD⟨SENTINEL_ID_2⟩EEFF...⟨SENTINEL_ID_3⟩GKGK⟨EOS⟩
AbAg Bind	⟨@TOKENIZER-TYPE=AA⟩ ⟨BINDING_AFFINITY_CLASS⟩⟨SENTINEL_ID_0⟩⟨@TOKENIZER-TYPE=AA@MAX-LEN=700⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_ANTIBODY_HEAVY_CHAIN⟩ EVQ...KSC ⟨@TOKENIZER-TYPE=AA@MAX-LEN=700⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_ANTIGEN⟩ MEL...YEG ⟨EOS⟩	⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩
TCR Bind	⟨@TOKENIZER-TYPE=AA⟩ ⟨BINDING_AFFINITY_CLASS⟩⟨SENTINEL_ID_0⟩⟨@TOKENIZER-TYPE=AA@MAX-LEN=700⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_TCR_BETA_VDJ⟩ TIQ...TVV ⟨@TOKENIZER-TYPE=AA@MAX-LEN=170⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_EPITOPE⟩ LEPLVDLPI ⟨EOS⟩	⟨@TOKENIZER-TYPE=AA⟩⟨SENTINEL_ID_0⟩⟨1⟩⟨EOS⟩
PPI $\Delta\Delta G$	⟨@TOKENIZER-TYPE=AA⟩⟨GENERAL_AFFINITY_CLASS⟩⟨@TOKENIZER-TYPE=SCALARS_LITERALS⟩<MASK>⟨@TOKENIZER-TYPE=AA⟩⟨COMPLEX_ENTITY⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_GENERAL_PROTEIN⟩IS...VY⟨@TOKENIZER-TYPE=AA⟩⟨COMPLEX_ENTITY⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_GENERAL_PROTEIN⟩DC...KCNF ...KC⟨@TOKENIZER-TYPE=AA⟩⟨MUTATED⟩⟨MOLECULAR_ENTITY⟩⟨MOLECULAR_ENTITY_GENERAL_PROTEIN⟩DC...KCQF ...KC ⟨EOS⟩	⟨@TOKENIZER-TYPE=AA⟩⟨GENERAL_AFFINITY_CLASS⟩⟨@TOKENIZER-TYPE=SCALARS_LITERALS⟩0.244⟨@TOKENIZER-TYPE=AA⟩⟨COMPLEX_ENTITY⟩ ...KC⟨EOS⟩

Continued on next page

Table S1 – continued from previous page

Bench.	Encoder input	Decoder label
DTI	⟨@TOKENIZER-TYPE=AA⟩<MASK> ⟨@TOKENIZER-TYPE=AA⟩⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY_GENERAL_PROTEIN⟩ **CC=..**⟨@TOKENIZER-TYPE=SMILES⟩ ⟨MOLECULAR_ENTITY⟩ ⟨MOLECULAR_ENTITY _SMALL_MOLECULE⟩ **AD..** ⟨EOS⟩	⟨@TOKENIZER-TYPE=SCALARS _LITERALS⟩ {standardized pKd}

Table S2 Cell type additional results

model	Accuracy	F1	Precision	Recall
scBERT	0.759	0.691	N/A	N/A
CIForm	0.820	0.710	N/A	N/A
MAMMAL	**0.856±0.004**	**0.763±0.012**	0.774±0.016	0.761±0.011

Table S3 Statistics of GDSC Datasets

Dataset	# Cell lines	# Drugs	# Cell-Drug pairs
Cancer-Drug Resp.1	958	208	177K
Cancer-Drug Resp.2	805	137	92K
Cancer-Drug Resp.3	561	223	107K

Table S4 Ab Infilling additional results

model	CDRH1-AAR	CDRH2-AAR	CDRH3-AAR	CDRL1-AAR	CDRL2-AAR	CDRL3-AAR
dyMEAN [32]	0.757	0.685	0.375	0.755	0.831	0.521
MAMMAL	**0.832 ±0.003**	**0.742 ±0.012**	**0.446 ±0.002**	**0.780 ±0.017**	**0.844 ±0.012**	**0.724 ±0.010**