Screening Fist

James Engleback

June 24, 2022

Contents

1 Abstract

Screening Fist is a screening operation used to train a machine learning model to predict the likelihood of a given enzyme binding to a given small molecule from sequence. The model is pre-trained on a large and general dataset of protein and small molecule binding pairs scraped from several online sources before retraining on the screening data. The screening data is generated in-house with a high-throughput, custom developed Cytochrome P450-small molecule binding assay, with which five P450 BM3 mutants were each screened for binding against 980 drug-like molecules.

The retrained model can be used for virtual screening of new enzyme sequences against a specific target molecule, or to query a single sequence against many prospective small molecule binding partners. It can also be used to design subsequent rounds of screening based on the expected information gain of an experimental design.

2 Introduction

2.1 Engineering Problem and Context

Herbicide-resistant crops are important for global agriculture because they mitigate yield losses due to weeds and give farmers extra flexibility in their herbicide application programs, which is important to suppress emergence of herbicide-resistant weeds.

Herbicides kill plants by inhibiting key metabolic processes and their species-specificity is determined by susceptibility of herbicide target and their ability to metabolize the herbicide.

Figure 1: The HPPD-inhibiting herbicide mesotrione and its primary metabolite 5-hydroxy mesotrione - hydroxylated at the C_5 position naturally in some plants by indigenous Cytochrome P450s, priming it for subsequent stages of detoxification.

mesotrione   metabolite
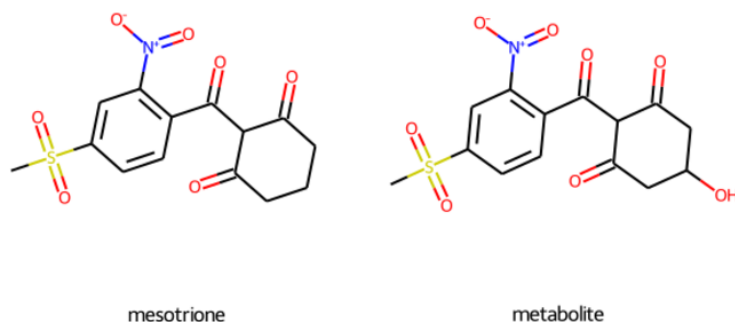
HPPD inhibitors are a key herbicide class that cause leaf bleaching and death in susceptible plants. HPPD inhibition disrupts tyrosine catabolism which disrupts UV-protection via carotenoid production and photosynthetic electron shuttling via plastoquinone, leading to death by UV damage and radical toxicity.
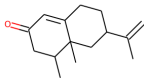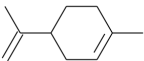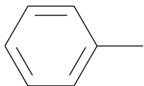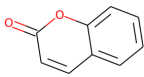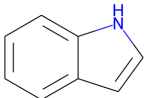
Engineering HPPD-inhibitor resistance into plants have used the HPPD and metabolic enzymes from naturally resistant species like *Avena fatua*. **note: ref this.** CYP72A1 is a cytochrome P450 that initiates metabolism of HPPD herbicides by ring hydroxylation at C_5 and has been explored as a means of engineering HPPD herbicide metabolism into soybean. **Figure 1** shows the chemical structure of mesotrione and its primary metabolite 5-hydroxy mesotrione.

Cytochrome P450s are a ubiquitous class of heme-dependent oxidoreductases that are frequently involved in xenobiotic metabolism. Bacterial P450s have been engineered to catalyse a range of xenobiotic biotransformations. The bacterial P450 BM3 from *Bacillus megaterium* is one such bacterial P450 whose structure has been studied extensively [1].

Notably, BM3 is a popular target for enzyme engineering studies, owing in part to its extensive documentation, ease of expression and *evolvability* [2]. Some examples of BM3 mutants with altered substrate scope are shown in **Table 4** [3].

The BM3 A82F/F87V mutant has a broad substrate specificity [4], and represents an attractive starting point for engineering altered substrate scope, however it has no binding activity towards the HPPD herbicide mesotrione.

Table 1: Examples of engineered BM3 Mutants with altered substrate scope [3]

Mutant	Substrate scope	Structure
F87A/I263A/(A328I)	Nootkatone	
A328F	Limonene	
R47L/Y51F	Alkylbenzenes (no heteroatoms)	
R47L/F87V/L188Q (LVQ)	Coumarins	
A74G/F87V/L188Q (GVQ)	Indole, Lovastatinn, beta-ionine, organophosphates, polyaromatic hydrocarbons, chlorinated dioxins	

2.2 Technologies Used

2.2.1 Machine Learning

2.2.2 Transfer Learning

Transfer learning is a popular approach to enhancing model accuracy and involves pretraining the model on a large, general dataset before transferring to a new task. The effect is a reduction in the number of samples required to train the model. Pre-training datasets do no necessarily need to be closely related to the target task, for example: models initially trained on the CIFAR 100 dataset (containing cars, animals etc. has been successfully re-tasked on phenotype identification from microscopy images [5].

In the domain of protein sequence-based machine learning, thoroughly pre-trained models are available for generating a neural embedding of a given protein that can improve sample efficiency in downstream learning tasks. Generally, these models are large attention-based models trained *unsupervised* on a large corpus of protein sequences, like the *TrEMBL* collection of *Uniprot*. In this case, unsupervised training often entails reconstruction of a distorted or masked protein sequence and is run on hardware far beyond the budget of this project.

2.2.3 The TAPE Benchmark

note: explain relevance, flesh out

Tasks Assessing Protein Embeddings [6] is a benchmark for comparing numerical representations of protein sequence (learned or otherwise) on a set of biological learning tasks from different domains of protein science. It currently contains five tasks:

1. **Secondary Structure Prediction Task:**
2. **Structural Contact Prediction Task:**
3. **Remote Homology Detection:**
4. **Fluorescent Protein Landscape Prediction:**
5. **Protein Stability Landscape Prediction:**

Tasks 4 and 5 are most applicable to protein engineering, since they involve metric prediction from a set of largely similar protein sequences. The leader boards for performance on these two tasks as of 5 Jun 2022 are:

Fluorescence:

Stability:

Facebook AI Research - Evolutionary-Scale Modelling note: explain model architecture and purpose, set of pretrained models and availability

2.3 Related Work

note - do:

Table 2: Fluorescence TAPE benchmark leader boards

Ranking	Model	Spearman's rho
1.	Transformer	0.68
2.	LSTM	0.67
2.	Unirep	0.67
4.	Bepler	0.33
5.	ResNet	0.21
6.	One Hot	0.14

Table 3: Stability TAPE benchmark leader boards

Ranking	Model	Spearman's rho
1.	Transformer	0.73
1.	Unirep	0.73
1.	ResNet	0.73
4.	LSTM	0.69
5.	Bepler	0.64
6.	One Hot	0.19

- **biswas**

2.4 Overview of This Work

This project is a proof of concept for a new approach towards enzyme engineering using machine learning. Broadly speaking, it involves screening a number of enzyme mutants for activity towards a large number of compounds of interest. This data is then used to train a machine learning model to predict the likelihood of activity between a given protein sequence and small molecule structure. The trained model has two applications:

1. Virtually screen a large number of potential protein and small molecule candidates for potential activity. These screens can be used to explore potential substrate scope of a given protein, or as a virtual fitness function for use in virtual directed evolution - where a sequence optimization algorithm evaluates sequences against a fitness function.

2. Optimal design of subsequent screening rounds based on the expected information gain (EIG) of a candidate experimental design.

The specific approach in this project is to screen a local area of sequence space surrounding the wild-type P450 BM3 against a library of 980 FDA-approved drug compounds. Drugs share chemical properties with herbicides so are suitable for a proof of concept study.

3 Methods and Development

3.1 Assay Development

In order to generate a P450 BM3-specific dataset on which a model could be re-trained to make binding likelihood predictions on drug-like molecules, a high throughput P450 binding detection assay was developed. The assay is based on traditional UV-Visible light spectroscopy-based techniques for detection of P450-ligand binding, miniaturized into a 384 well format. It relies heavily on automation and a throughput of 980 compounds per day is demonstrated.

3.1.1 Aim

The initial aims of this development work were:

- Develop a high throughput P450-ligand binding assay based on established biophysical characterisation techniques.
- Develop necessary software for design and analysis of each assay.
- Compare the precision and accuracy of the assay to existing techniques.

3.1.2 Basis: UV-Visible Spectroscopy for Monitoring Cytochrome P450-Ligand Binding

The assay is based on a technique for quantifying P450-ligand interactions based on UV-visible photospectroscopy. The technique consists of the purified Cytochrome P450 heme domain in question in a neutral buffer at around 5-10 µM in a optically clear cuvette. Since only the heme-containing domain of the P450 is used, no chemical reactions are expected to take place which removes time-sensitivity from the assay.

The UV-visible light absorbance of the sample is typically measured for all wavelengths between 200 and 800 nm, which for a P450 without a ligand bound in the active site should show a large and defined absorbance peak at around 420 nm.

After an initial absorbance measurement of the ligind-free P450, the compound of interest can be titrated into the sample. On binding to the ligand, the absorbance profile of the P450 changes such that the absorbance at 420 nm (A_{420}) decreases and absorbance at 390 nm (A_{390}) increases.

The change in A_{420} and A_{390} in response to change in ligand concentration can be quantified and used to derive metrics that indicate affinity between the ligand and P450 using Michaelis-Menten kinetics models.

The original Michaelis-Menten model of enzyme kinetics states:

$$v = V_{max} \frac{[S]}{[S] + K_M}$$

where v is the reaction velocity - the rate of an enzymatic reaction. V_{max} is the maximum possible v for this particular enzyme-substrate pair, $[S]$ is the concentration of the substrate and K_M is the $[S]$ at which $v = V_{max}$.

V_{max} and K_M are useful metrics for quantifying the binding interaction between enzyme and substrate, where low K_M indicates a tight binding interaction and a high V_{max} indicates a large magnitude of response.

Important assumptions in the Michaelis-Menten model of kinetics are:

1. The concentration of enzyme is $< K_d$

2. The rate of reaction is directly proportional to the concentration of the substituents

3. The reaction is at chemical equilibrium at the time of measurement

4. The interaction is reversible

A variant of this model is applied to Cytochrome P450 photospectroscopy assays, where the response to a ligand is detectable in the 390-420 nm region. Ligand binding to a P450 alters the electron environment in the heme, which changes the UV-Visible light absorbance profile. Specifically, absorbance at 420 nm is reduced and absorbance at 390 nm increases.

In this case, v is substituted for $\Delta A_{390} - \Delta A_{420}$ - the magnitude of the P450 response and K_M is substituted for K_d - the dissociation constant between the enzyme and ligand. This yields the formula:

$$Response = \Delta A_{390} - \Delta A_{420} = V_{max}\frac{[S]}{[S] + K_d}$$

An example of this is shown in **figure 2**, where the fatty acid arachadionic acid was titrated into a sample of P450 BM3 wild-type.

3.1.3 Development

This style of assay was miniaturized into a 384-well format for the purpose of this project. The 384-well format permits high throughput screening of compounds for binding with a given P450 provided it is sufficiently stable to last the duration of the experiment without degrading and interfering with measurement.

At micro-scales, precision dispensing of both compound and protein is critical for overall assay precision. To achieve sufficient precision, compounds were dispensed using an *Echo 550* acoustic liquid handling device, which uses ultrasound to move liquid from a source plate to a destination plate. In order to achieve the desired dispensing pattern, a custom python package, `echo`, was developed. `echo` facilitates dispensing of compounds in the gradients required by output of a picklist `csv` file that can be used directly in the *Echo*. A critical function of this package is that it tracks the volume of compounds in the source plate, which is important in the common case that one compound must be spread over multiple wells. The package was indispensable in this work.

During early stages of development, an automated bulk liquid dispensing device - a *ThermodropMulti* was available. The device uses precise peristaltic pumps to dispense a single liquid into a plate at a single volume very quickly. Precision in the bulk fluid volume in the plate is helpful in downstream analysis due to the consistent path lengths and concentrations in

Figure 2: Response of P450 BM3 to Arachadionic Acid from a titration experiment, from which steady-state Michaelis-Menten kinetics can be derived. The legend shows the concentration of arachadionic acid in µM.
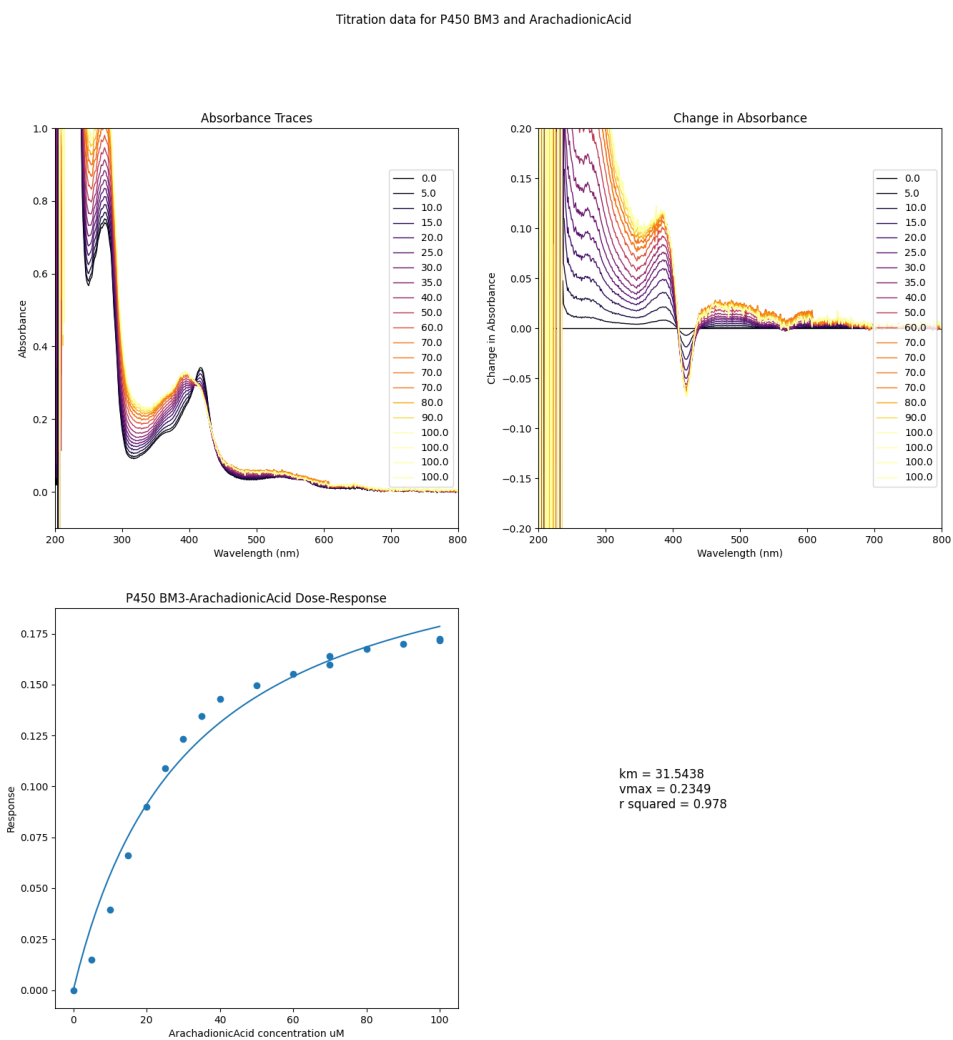
each well, something that was found to be less consistent when using a multichannel pipette to achieve the same task. **Figure 3** shows an example of both the *ThermodropMulti* and *Echo* being used to capture K_d and V_{max} for the binding interaction between P450 BM3 wild-type and arachadionic acid.

3.2   Assay Protocol

This is a assay protocol for detecting binding interactions between a Cytochrome P450 and multiple small molecule compounds. The assay has demonstrated scale to a library of 980 compounds and five P450 mutants and with some small adjustments could be improved in scale and precision.

It works in 384 well microplates and uses a microplate reader to capture absorbance profiles from 220-800 nm wavelengths, from which a pattern associated with a P450-small molecule binding interactions can be detected and quantified.

It was designed for profiling and modelling the effect of mutations on a P450's substrate binding preferences. This was tested with five P450 mutants against 980 drug-like compounds. It requires purified P450 protein which limits the rate of data generation, though can scale to more compounds.

3.2.1   Requirements

Essential:

- **Hardware:**

    – **Microplate reader:** Able to read absorbance for all wavelengths between 220 and 800 nm. Used here: *BMG ClarioStar and FluoStar* microplate readers.

    – **Labcyte Echo [500 550]:** Acoustic liquid handlers for precise compound dispensing.

    – **Consumables:** In absence of a high precision liquid handling machine, serial dilution of compounds would probably be fine.

    – **Enzyme:** A purified Cytochrome P450 - used here were mutants of P450 BM3 at 800 µM. Note that BM3 is fairly stable at room temperature which facilitates queuing large batches of plates to the plate reader. You could run the assay at a low temperature if you use a solvent other than DMSO, which freezes at 19ᵒC, which interferes with measurement.

    – **Compound Library:** A large number of compounds in solvent (e.g. DMSO) in a microplate format. Used here was a 980 compound library, dissolved at 10 mM in DMSO in 96 well format.

    – **Buffer:** must be optically clear the protein must be stable in it. Must not contain potential ligands. Used here was 100 mM Potassium Phosphate at pH 7.0 - chosen based on traditional wisdom.

Figure 3: Michaelis-Menten steady-state kinetics of arachadionic acid and wild-type P450 BM3 captured during assay development. Notably, in this experiment both an *Echo* and *ThermodropMulti* were used in dispensing compound and protein respectively, which accounts for the neatness. The traces were smoothed using a Gaussian kernel to account for noise.
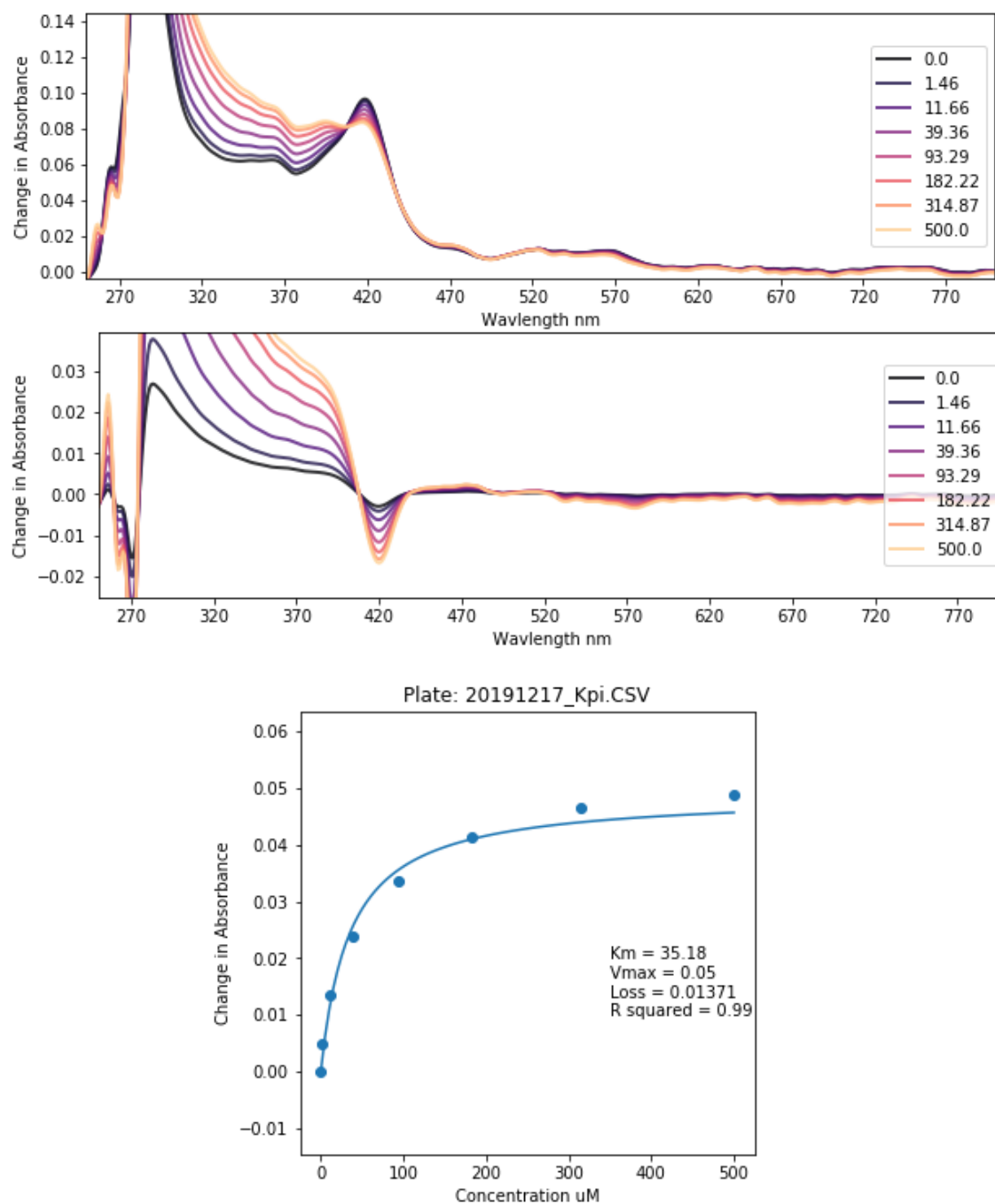
– **384 well microplates - clear bottom:** Assay plates with at least 30 µl working volume. Some treated surfaces may be more suitable for unstable proteins. Ideally have minimal absorbance in the 390-420 nm region, but this can be corrected for with controls. Used here: *ThermoFisher Nunc 384 Well Plates* - **384 well *Labcyte Echo* source plates:** for dispensing compounds to assay plates. Used here were the *Low Dead Volume (LDV)* variety, which have a working volumne of 2.5-12.5 µl, which limits compound waste compared to the standard *Echo* plates (12-65 µl).

Optional:

- **Hardware:**

  – **Bulk liquid dispensing:** can be far more accurate than a multichannel pipette when dispensing protein or buffer into wells. During development, both a *ThermodropMulti* peristaltic dispenser and a *Hamilton Star* liquid handling robot. Both work well, though use of a bulk liquid dispensing machine is recommended given their speed, lower unit cost and lack of requirement for pipette tips.

  – **Microplate reader plate loader:** Autoloading plates into the reader increases throughput capacity significantly. I used a **BMG ClarioStar** plate reader with a stacker module.

- **Consumables:**

  – **BSA:** in assay buffer may have a stabilizing effect on the enzyme - which would improve time stability and reduce errors. Time stability is important for scalability.

  – **384 well Labcyte Echo DMSO Trays:** for control for DMSO concentration in assay wells by topping up each assay plate to a fixed concentration. Around 5% is ok with BM3.

3.2.2 Procedure

Summary:

1. **Design *Echo* picklists:**

   - An *Echo* can accept a `csv` file with column headers: `Transfer Volume`, `Volume`, `Destination Well`, `Source Well` and optionally: `Destination Plate Name` and `Source Plate Name`. The Volume must be in nano litres and a multiple of 2.5 and the Source and Destination wells must be in the format `[A-Z][0-9]+` and exist in the plate layout specified to the *Echo* client at runtime.

   - The picklist(s) can be generated in a spreadsheet exported to `.csv` or programmatically. Documentation for the `python` tools used are [documented here.](picklists.md)

2. **Dispense compounds into *Echo* source plates** This can be done with a multichannel pipette, and requires one tip per compound. If the total volume of each compound required is greater than 60 µl then a standard polypropylene *Echo* plate should be used, otherwise a low dead volume plate may be economical If not, or for valuable compounds, Low Dead Volume *Echo* may plates should be used. These have a working

volume of 2.5-12.5 µl, outside of which the *Echo* will refuse to dispense. You may need to dispense the same compounds into multiple source wells and the picklists must be designed accordingly.

3. **Dispense compounds from *Echo* source plates to empty assay plates**

   (a) Transfer the picklist `.csv` to the *Echo* host computer.

   (b) Launch the *Echo Plate Reformat* client there:

   (c) Create New Protocol

   (d) Select Custom Region Definitions

   (e) `File` > `Import Regions` and select your picklist `.csv`

   (f) **Optional:** Specify the log output in `Options`, simulate with `Run` > `Simulate`

   (g) Save and run, optionally simulating the run first. Multiple copies of a set of destination plates can be specified if the source plates contain sufficient compound volume.

4. **Stopping point:** Length of pause depends on rate of DMSO evaporation from destination/assay plates and the stability of the compounds at the plate storage conditions. Plates stored in a stack should limit evaporation rate to an extent, though specialised lids for *Echo* plates that limit DMSO evaporation are available. Up to 24 hours seems ok.

5. **Thaw purified P450, make stock of 10 µM in a neutral buffer, enough for 15.36+ ml per plate (40 µl per well)** I heard that thawing fast limits ice crystal formation, which could destroy some protein. Optionally, in a microcetrifuge, pre-cooled to 4C, spin the protein at 14,000 rpm and carefully transferr to fresh tubes to remove unfolded protein.

   (a) Measure the stock concentration of the protein in a UV-Vis spectrometer by taking an absorbance trace from 200-800 nm, diluted in the destination buffer. There should be a peak at 420 nm, the height of which can be used to calculate the protein concentration with the following equation:

   $$[P450] = ael$$

   where a is absorbance and e is the extiction coefficient - 95 for P450 BM3 heme domain. l is the path length in cm and for the cuvettes used was 1. Use the measured stock concentration of P450 to create a working stock of around 10 µM. 10µM was chosen because it yeilds a reasonably strong signal when in plates. Varying the protein concentration doesn't have a big effect on measurements, so err towards using more.

   (b) Dilute in neutral buffer to the target working concentration. Filtration through a 22 µm syringe filter can remove some precipitates. Vacuum filtration can work too but in practice, the protein can pick up binding partners from the filtration equipment contaminants, which can ruin downstream measurements.

6. **Dispense the diluted protein into the assay plates, centrifuge**

   An electric multichannel pipette works but accuracy is more limited than with automated dispensing. 38 μl of protein working stock needs to be dispensed into each well, which brings the total well volume to 40 μl in cases where the volume of compounds in DMSO in each well is 2 ul. If the volume of DMSO in destination wells is not a constant 2 ul, then default to 38 μl of the protein stock. The variation in total volume can be corrected for in compound concentration calculations, though the path lenght will vary which affects precision.

   Better than that is a precise bulk liquid handling device. I used a *ThermodropMulti* for a while which was fast and accurate. Occasionally a nozzle would become blocked either with DMSO ice or precipitates, though the protein still dispensed into the correct well. Blockages can be cleared by disassembling the pump head, coupling a syringe of water to the nozzle and flushing.

   It may be necessary to dispense some control plates, with everything but protein. This is useful to correct for the intrinsic absorbance of the plate and buffer, as well as the compounds themselves which sometimes have absorbance at the measurement wavelengths. A control set of plates for every protein screen may be unnecessary and expensive. One good one should be ok.

   Centrifuge the plates for 2-5 mins at around 2000 rpm to push the well contents to the bottom. This step can also ensure that meniscuses are flat and uniform and remove bubbles. If possible, centrifuge at room temperature to avoid DMSO ice formation.

7. **Capture UV-Visible light absorbance data between 220 and 800 nm from plates in a microplate reader at room temperature within 3 hours:**

   The protein is fairly stable over the course of 3 hours. On a BMG platereader, measurements take about 15 minutes per plate including the data transfer from device to host machine. Using an automated plate loader is recommended, for example a BMG Stacker module. In that case, put an empty or waste plate on the top of the stack to limit evaporation from the top assay plate. The BMG ClarioStar can be instructed not to read the last plate.

   The stacker occasionally jams due to a solenoid error, which can be due to a misaligned stack of plates. It is advisable to un-stack and re-stack the plates using the stacker to check for this kind of issue prior to measurement.

8. **Data analysis overview**

   (a) Export the plate measurement data to a workable format, like `.csv`. In the BMG Mars software, the operation is simple but on all host machines I've tried it on have been unreasonably slow to open the data files prior to export.

   (b) Index the files to their experiments. I used a `config.yml` file to track this.

   (c) **Analysis**

      i. Match compounds to plate well data.

ii. Match the *Echo* exceptions report to wells to find the actual compound volume in each well.

iii. From each trace, subtract its own absorbance A_{800} at 800 nm. This accounts for baseline drift which can be caused by light scattering from precipitates.

iv. If correcting for compound absorbance with control plates, then subtract the absorbance of each test well from each control well. If the actual compound volumes of the test and control don't match up, it can be an issue if the compound interferes with the absorbance in the 390-420 nm region. If the compound absorbance changes predictably then it can be interpolated.

v. Curves can be smoothed with Gaussian smoothing using `scipy.ndimage.gaussian_filter1d` if necessary. Sources of a jagged curve can be precipitates, which can interfere with downstream analysis.

vi. At this point, changes in the P450 absorbance trace can be identified. Common categories of trace are:

  • Clean absorbance trace, no shift.

  • Clean absorbance trace, peak shift from 420 to 390 nm.

  • Clean absorbance trace, peak shift from 420 to 424 nm.

  • Compound interference in absorbance trace.

  • Light scattering in absorbance trace.

vii. For clean traces with a peak shift from 420 to 390 or 424 nm, biding response can be calculated using the $|\Delta A390| - |\Delta A_{420}|$ or $|\Delta A420| - |\Delta A_{420}|$ for each compound concentration. With a set of concentration-response data points, the binding dissociation constant K_d can be calculated using the Michaelis-Menten equation for enzyme kinetics:

$$Response = \frac{V_{max} \times [S]}{K_d + [S]}$$

$$Response = |\Delta A_{390}| - |\Delta A_{420}|$$

Where $[S]$ is a given substrate concentration and V_{max} is the maximum response magnitude possible from the P450 being tested from this compound. The metrics K_d and V_{max} can be derived by fitting $|\Delta A390| - |\Delta A_{420}| = \frac{V_{max} \times [S]}{K_d + [S]}$ can be fit to the P450 substrate concentration-response data points using a curve fit algorithm like `scipy.optimize.curve_fit`.

Useful additional metrics for each compound are R^2 score of the curve fit, a data quality metric.

An ideal end output of this analysis as a table of compounds, P450s and a qualification or quantification of their binding interactions.

Table 4: BM3 mutants used in screening

ID	Mutations	PDB
WT		1BU7
A82F	A82F	4KEW
DM	A82F/F87V	4KEY
1YQO	T268A	1YQO
1YQP	T268N	1YQP

3.3 Enzyme Production

3.3.1 Summary

This page contains the methods for producing the enzymes used in this screening program. The enzymes are variants of the Cytochrome P450 BM3:

The page shows the method used to create the mutant BM3 expression plasmid DNA, expression of the mutants in *E. coli* and their purification.

3.3.2 Aims

- Create expression plasmids containing the target mutants from an in-house starting point - `bm3-wt.gb`.

- Sequence the plasmids to confirm they carry the mutations

- Express the mutants in *E. coli* using those plasmids.

- Purify the mutant protein from the *E. coli* harvest.

3.3.3 DNA

3.3.4 Starting Material

An heirloom BM3 Wild-type (heme domain) expression plasmid, `bm3-wt.gb`, was inherited and used as the basis for DNA work in this project. The plasmid is a pET14 expression vector where the BM3 gene has a 6xHis purification tag at the N-terminus, flanked by a T7 promoter and terminator which leads to high yields in strains of *E. coli* containing the T7 RNA polymerase. The plasmid also encodes ampicillin resistance and a *ColE1* replication origin which leads to a low copy number.

3.3.5 Primer design and Acquisition

Mutations were introduced to the wild-type sequence via Polymerase Chain Reaction (PCR)-based site-directed mutagenesis. Two methods were considered for this task based on commercially available kits, where each imposes different constraints on primer design. Efforts were made to automate primer design as far as possible with scalability in mind.

The PCR kits used were:

1. *New England Biolabs (NEB) Q5 mutagenesis kit* - which requires that primers facilitate cloning of a linear DNA strand from the circular template plasmid and mutation payloads are carried in the tail of one primer. The kit includes a cocktail of the DNAse *DPN1*, which disassembles template plasmid methylated in *E. coli* and a kinase and ligase that work to join the ends of the linear DNA into a circular plasmid. The reaction is restricted to one payload.

2. *Agilent Quickchange mutagenesis kit* - which requires a pair of overlapping primers that carry the mutation payload in the mid-section. This cloning method produces circular DNA carrying the targeted changes. It has the advantage of allowing multiple payloads carried by multiple primer sets.

Two important considerations based on the template sequence are:

1. Adenine-Thymine (AT) richness of the template sequence. Compared to cytosine and guanine (C and G), A and T bind to their complimentary bases weakly. This results in weak primer binding to the template sequence, measurable by a low primer *melting temperature* T_m. To compensate, primers must be longer than they otherwise would be for a sequence richer in CG, which increases their cost and their chance of self-binding. The template sequence used here is AT-rich - at 42%.

2. Repetitions and palindromic regions of the template sequence. If the sequence surrounding a mutation target area contains these features, then the likelihood of *mis-priming* by binding to an off-target sequence area is high, so too is the likelihood of a non-functional, self-binding primer.

3.3.6   PCR and Work Up

3.3.7   Sequencing

Purified plasmid DNA ostensibly conataining the target mutations, having been harvested and purified from DH5a *E. coli* cells, was shipped to *Eurofins Genomics* for sequencing using their *TubeSeq* service, which uses a variant of Sanger Sequencing. Sequencing primers for this matched the T7 promoter and terminator and provided coverage of the targetted region.

note: include sequencing, flesh out

3.4   Protein Expression

Having been sequenced and confirmed to carry the target mutations, the mutant plasmids were used to produce the mutant protein *en masse* via a *BL21 DE3 E. coli* strain, which contains a T7 RNA polymerase under the control of a *lac* promoter.

BM3 mutants were expressed in *NEB* BL21 (DE3) *E. coli* cells grown in *Formedium* auto-induction TB media at 25°C for 24 hours with shaking at 180 rpm. Cells were harvested by centrifugation in a *Beckman* JA16 rotor at 6000 rpm for 10 minutes and stored at -20 °C. Variants were purified from the cells by nickel-affinity chromatography, dialysis into buffer 1, concentration to 500-1000 μM and storage at -80 °C.

3.4.1 Materials

- Expression plasmid encoding mutant P450 BM3

- *BL21 DE3 E. coli* - NEB. This domesticated *E.coli* strain is shipped in a transformation buffer.

- Auto-induction *Terrific Broth* (TB) media, which contains glucose and a lactose analog. The lactose analog triggers expression of T7 RNA-polymerase in *BL21 DE3 E. coli* and the subsequent expression of the target protein between the T7 promoter and terminator regions. The glucose inhibits this until it is consumed by the cells, which allows them to multiply to sufficient numbers before diverting energy to production of the target protein.

- Ampicillin - the antibiotic for which resistance is encoded in the target plasmid, ensuring that all cells in the growth media contain this resistance. Assuming no ampicillin-resistant contaminants, all cells should be *BL21 DE3 E. coli* containing the target plasmid.

- Δ Amino-Levulnic acid (Δ-ALA) - a precursor to heme, ensuring heme availability for the large amount of BM3.

3.4.2 Method

3.4.3 Purification

3.5 Screening

Throughput the screening process, the protocol described in section 3.2 was employed. Six concentrations of each target compound were dispensed with the aim of determination of the V_{max} and K_d: 0.0, 5.26, 55.61, 156.72, 306.74 and 500 μM. These values were generated from six linearly spaced values between 0 and 1 squared, then scaled to the starting concentration and final volume. This method saved some compound compared to a straightforward linear spacing of values between 0 and 500 μM - the maximum concentration possible given a 10 mM starting stock in DMSO without exceeding a total of 5% v/v solvent.

During the first run, control experiments where compounds were present in buffer only were included for downstream correction of the compounds inherent UV-Visible light absorbance, though it was thought to be unnecessary to repeat this for each subsequent screen, since it is wasteful of resources, particularly the compounds themselves.

Each of the five enzymes was screened against all 980 compounds over the course of a day each.

3.6 Model Design and Construction

The model aims to predict the binding likelihood between a given protein sequence and chemical structure, illustrated in equation 3.6.

$$P_{binding} = fn(sequence, smiles)$$

This page describes the deep learning model constructed for this project. The model is designed to estimate the likelihood of a binding interaction between a given Cytochrome P450 sequence and a ligand SMILES code. The intended end uses of the model are:

1. Virtually screening sequences for potential activity with a given compound. 2. Optimally design an enzyme-ligand screening experiment.

3.6.1   Approach: Recommender Systems

Abstractly, the problem of predicting the binding likelihood between a one of m proteins and one of n small molecules can be likened to filling empty the values of an $n \times m$ matrix, where rows and columns refer to proteins and small molecules and values are the probability of a binding interaction between the two:

$$
\begin{array}{cccc}
& compound_i & compound_{i+1} & ... & compound_{i+n} \\
sequence_i & P_{binding_{i,j}} & & & \\
sequence_{i+1} & & & & \\
\vdots & & & & \ddots \\
sequence_{i+m} & & & &
\end{array}
$$

Some $P_{binding}$ values are known, which in the perspective of $n \times m$ possible values where n and m approach infinity, coverage is sparse.

This type of problem has been addressed in recommender systems, which in the context of streaming services translates to a matrix of n users and m paces of content. Known values are likes and engagement metrics and are similarly sparse, and blanks can be filled with the probability of a successful recommendation.

Machine learning models can be trained to predict the unknown values based on a numerical representation of the user and content. The prediction can be cast as a classification problem. To overcome the lack of negative data points, presumed negative data can be generated by sampling a random user: content pair, which should be treated with caution.

In this work, a machine learning model classifies pairs of protein sequence and small molecules as binding or not. Negative samples are generated by randomly sampling a sequence and small molecule, which given the vastness of sequence and chemical space may be reasonable in a large number of cases, though this assumption is treated with caution. A Binary Cross Entropy loss function is employed here where true positives and synthesized negatives are weighted evenly.

3.6.2   Data

Small molecules are represented as SMILES codes in the dataset, which are parsed using `rdkit` and then into 2048 bit fingerprint vectors using the `RdkitFingerprint`. Molecular fingerprints are generated by hashing functions based on an input molecule such that similar molecules are assigned similar fingerprints, which makes them useful in featurizing small molecules for machine learning tasks. The fingerprints are represented as a $b \times 2048$ tensor where b is batch size.

Sequences are represented in the dataset as strings where each character c_i is a single letter amino acid code:

$$c_i \subset ACDEFGHIKLMNPQRSTVWY$$

Roughly, characters are encoded as tensors of the integers that index their position in the list $ACDEFGHIKLMNPQRSTVWY$, with extra positions to represent null values, start of frame and end of frame characters.

3.6.3   Pre-Training Data

3.6.4   Training Data

3.6.5   Model Architecture

Given the model aims to predict:

$$P_{binding} = fn(sequence, smiles)$$

A neural network was devised to generate a prediction of $P_{binding}$ from an input of a protein sequence and a chemical SMILES code. The network can be split into three parts:

- **Sequence Embedding:** For a given protein $sequence$, output a tensor encoding a neural embedding $z_{sequence}$.
- **Chemical Embedding:** For a given chemical $smiles$ encoding, outputs an embedding z_{smiles}.
- **Prediction Head:** For the embeddings z_{smiles} and $z_{sequence}$, output a prediction $P_{binding}$.

The architecture is illustrated in 4.

3.6.6   Sequence Embedding

Although this model is the smallest of the ESM collection, on the single *NVIDIA Quadro RTX 6000* used it still occupied most of the 24 GB of available memory and most of the processing capability, which lead to long training times and difficulty in training more than one model in parallel on the same machine.

This could be remedied, however since in the complete `o3f.csv` dataset and the screening dataset there are 2947 unique sequences, so it was economical to pre-compute the embeddings and save them to disk. This resulted in a roughly 4× speedup in training time and massively reduced the memory requirements, allowing several models to be trained in parallel on a single GPU. This also saved costs significantly.

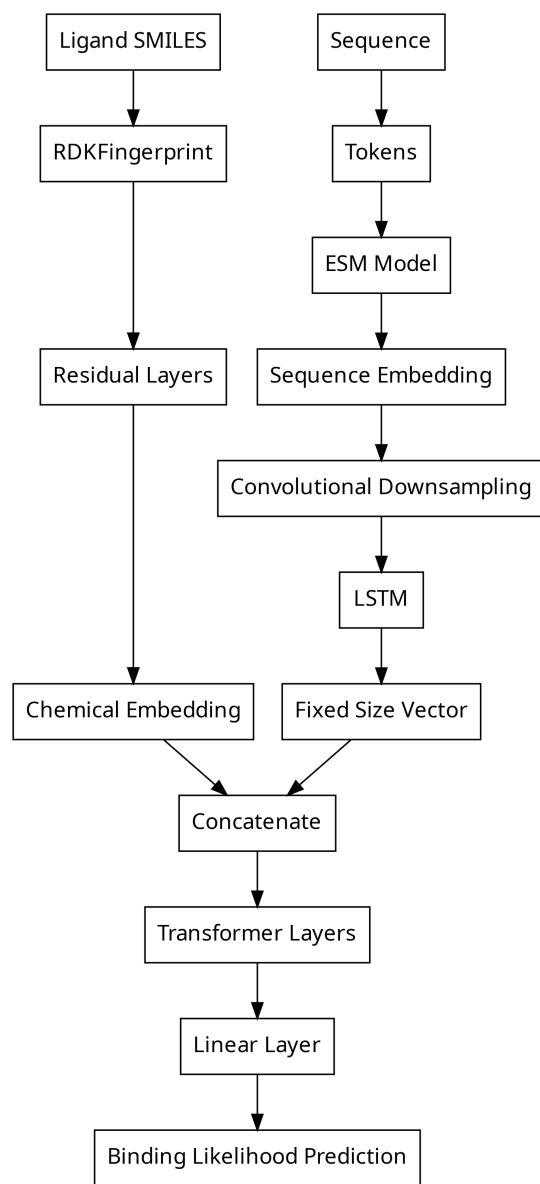Figure 4: Architecture of the model constructed for this project

Ligand SMILES	Sequence
↓	↓
RDKFingerprint	Tokens
	↓
	ESM Model
↓	↓
Residual Layers	Sequence Embedding
	↓
	Convolutional Downsampling
	↓
	LSTM
↓	↓
Chemical Embedding	Fixed Size Vector

Concatenate

↓

Transformer Layers

↓

Linear Layer

↓

Binding Likelihood Prediction

Table 5: Hardware specifications for machine used in model training.

Item	Specifications	Number	Size
CPU	Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz	8	
RAM	?	N/A	30 GB
Disk	?	1	630 GB
GPU	NVIDIA Quadro RTX 6000	1	20 GB VRAM

3.6.7 Chemical Embedding

As mentioned, chemical SMILES were hashed into chemical fingerprints using the `rdkit RDKFingerprint` method as a means of representation, yielding a 2048-bit vector for each compoundagainThe vectors were converted to tensors and served as an input to a residual neural network that output an embedding that would later be used to form a combined representation of both compound and sequence for binding likelihood prediction.

3.6.8 Prediction Head

The combined sequence and compound embeddings served as input to the prediction head, which output a single number that indicated a binding likelihood prediction for the two inputs.

Both residual neural networks and transformers were compared as prediction head architectures. Each consisted of 2-6 stacked layers of either residual or transformer layers with a fixed hidden layer size for convenience of automated assembly. The final layer in both cases was a single linear layer with a single output and a sigmoid function to output a number between 0 and 1 representing binding probability.

3.6.9 Pre-Training, Training and Evaluation

Training was done in two stages, each with a performance evaluation, during which several models with varying architectures and hyper-parameters were trained and compared. All training was done on a *Linode* `g1-gpu-rtx6000-1` which cost $1.50 per hour and was equipped with hardware specifications described in **table 5**.

1. **Pre-Training:** This was done with the larger, more general `o3f` dataset, which was randomly split into training and validation partitions, the latter of which was used sparingly to avoid model bias. Pre-training lasted up to 64 epochs with a batch size up to 64. For each sample, a random sequence and SMILES pair were sampled as a presumed negative sample. The loss function used was binary cross entropy used with an Adam (Adaptive momentum) optimizer. Loss was tracked live using the *Weights and Biases* API which was useful to evaluate models as they trained and terminate them where necessary. Model weights were saved in each epoch and after training the model was evaluated for precision and accuracy on a subset of the training data. The metrics gathered were:

(a) Mean binary cross entropy loss over evaluation.

(b) Mean precision

(c) A confusion matrix

(d) A receiver operator curve (ROC)

(e) A precision recall curve

(f) A detection error trade-off (DET) curve

2. **Training:** This was done with the manually annotated screening dataset. An issue with the data was the class imbalance in that there were very few positive examples relative to negative. This was addressed by using *Synthetic Minority Oversampling* (SMOTE) whereby the rarer positive data were re-sampled until they number that of the negative data. The total size of the re-sampled data was 6666 points, which were then split 3:1 into training and validation sets of size 4999 and 1667 respectively. A model pre-trained on the larger `o3f` dataset was re-trained on this set and evaluated for performance in the same manner as with the `o3f` data, visualised in the following section.

3.6.10 Evaluation

4 Results

4.1 Screening Data Analysis

The data gathered for the screening experiments was saved in the `git` repository `https://github.com/jamesengle` in the `lab` directory. Within the `lab` directory are directories for each screening experiment. These include a `config.yml` file which maps data files to experiments for analysis.

The script `sxfst/scripts/data_proc2.py` was used to process the data into a large `csv` file containing each raw trace and its metadata. By consolidating the data into a single file, fewer time-consuming operating system calls are made in downstream analysis which decreases the time cost of those tasks.

The script attempted to use the *Echo* logs from each run to calculate the actual volume of each compound dispensed, since towards the end of the screening program shortages of compound made it impossible to fully dispense some experiments. In retrospective this function did not perform properly, since all experiments are later annotated as having the target volumes of each compound. This cannot be the case since it was casually observed that some compounds were completely depleted by the end of the screening program. This issue is significant, leading to false negatives where a lack of response cannot be attributed to either a lack of compound or a true negative response. This coding issue can be solved, but was not due to time constraints.

The downstream analysis in question was done using the script `sxfst/scripts/data_analysis.py` which normalizes the traces, subtracts the control traces from them and calculates the P450 response to each compound, creating a plot in the process. An example of one such plot is in **figure 5**.

The traditional response calculation method using $\Delta A_{390} - \Delta A_{420}$ proved ineffective for analysis of this data due to the noise, which would often result in scattering that increased the absorbance at the short wavelengths in ways that were occasionally unpredictable.

Instead, the gradient of each trace was taken by convolution of the kernel $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$ across the smoothed and normalized traces. The gradients were invariant to total absorbance, which was often perturbed by what appeared to be light scattering, but achieved a similar effect to calculation of $\Delta A_{390} - \Delta A_{420}$. From the gradients, response was calculated as $|\frac{\delta y}{\delta x_{390}}| + |\frac{\delta y}{\delta x_{420}}|$.

Although this proved successful, the automated curve fitting to the response was not, outputting a constant value of 32 μM for each experiment - the lower K_d bound. With extra development this issue can likely be solved, but in order to keep within time constraints, an alternative solution was implemented.

The alternative solution was manual annotation of the plots output by the script. Though crude, this method did yield a list of ostensible *hits* as well as a list of anomalous results that were to be ignored.

Using this approach, 149 hits of the total 4900 compounds were identified, listed in the file `lab/hits.txt` in the repository. These were used to create a `csv` file containing the full sequence and SMILES code for each enzyme compound pair and a boolean value that was true if that experiment was annotated as a *hit*. There are several drawbacks to this that should be noted:

- **Human Error:** Although several passes over the data were done, the order was never randomized and there was only one annotator. Since there are 4900 experiments, even if the annotator averages 1 second per annotation the runtime of an annotation run is 81 minutes, which will likely cause fatigue and subsequent errors. Since the order was not randomized, fatigue-induced error is more likely to occur in the same regions of the dataset in each pass.

- **False Negatives** due to a lack of properly functional compound concentration calculation, which would yield results that appear negative due to a lack of compound without being properly labelled as such. Another source of false negatives is the obfuscation of hits by noise in the traces due to scattering or compound absorbance that in some cases proved hard to correct for.

4.2 Model Training

4.2.1 Pre-training

Given constraints on money towards computing resources, a hyper-parameter sweep was not viable. Instead, a small selection of models were trained and evaluated and the best of those was retrained on the screening data. Model architecture and hyper-parameters were directly configurable by supply of arguments to the training script `model/train.py`, from which the best performing model was obtained using the command `./train.py -i data/o3f.train.csv --transformer --cuda --emb_size_head 2560 --n_layers_head 3 --emb_size_fp`

Figure 5: A good example of a positive result or *hit* between BM3 A82F and *Gestodene*. K_d was calculated as 32 μM, which reflects an issue in the curve fitting rather than the actual K_d. **note - rescale axes, move kd text, legend title**
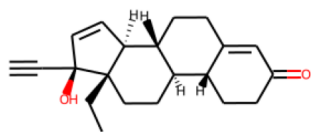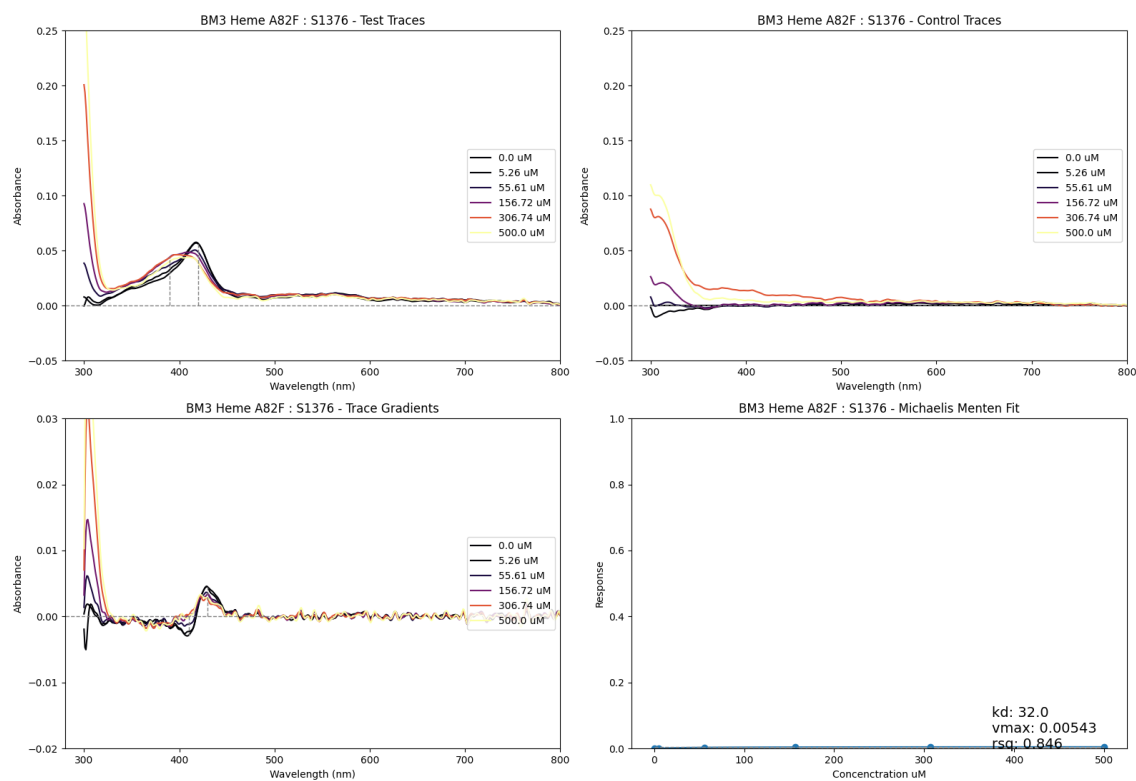
Table 6: Parameters for the best performing model in pre-training, designated the identifier *frosty-breeze-83* by the *Weights and Biases* API.

Parameter	Value
lr	10^{-5}
esm	esm1_t6_43M_UR50S
cuda	True
load	None
test	False
input	data/o3f.train.csv
epochs	320
batch_size	640
emb_size_fp	512
n_layers_fp	4
stride_pool	3
transformer	True
lstm_hs_pool	1024
emb_size_head	25
n_layers_head	3
kernel_size_pool	9
num_conv_layers_pool	3
num_lstm_layers_pool	2

```
512 --n_layers_fp 4 --num_conv_layers_pool 3 --kernel_size_pool 9 --stride_pool 3 --lstm_hs_pool
1024 --lr 1e-5 --batch_size 64 -e 32
```
, which is consolidated in **table 6**.

4.3   Model Application

5   Discussion and Future Work

- **Improvements (Issues):**
    - **Issues with screening library**
    - **Improvements to data processing**
    - **Improvements to screening dataset partitioning**
    - **Improvements to model application**

References

[1]   Andrew W Munro et al. "P450 BM3: the very model of a modern flavocytochrome". In: *Trends in biochemical sciences* 27.5 (2002), pp. 250–257.

[2]   Jesse D Bloom et al. "Protein stability promotes evolvability". In: *Proceedings of the National Academy of Sciences* 103.15 (2006), pp. 5869–5874.

Figure 6: Model evaluation report after pre-training for the model designated *frosty-breeze-83*. Evaluation was on `o3f.test.csv` - a held-back partition of the pre-training dataset. Depicted are: mean binary cross entropy loss the course of training on `o3f.train.csv`, a confusion matrix, a receiver operator curve (ROC), a precision recall curve, a detection error tradeoff (DET) curve and mean precision and mean binary cross-entropy loss over evaluation.
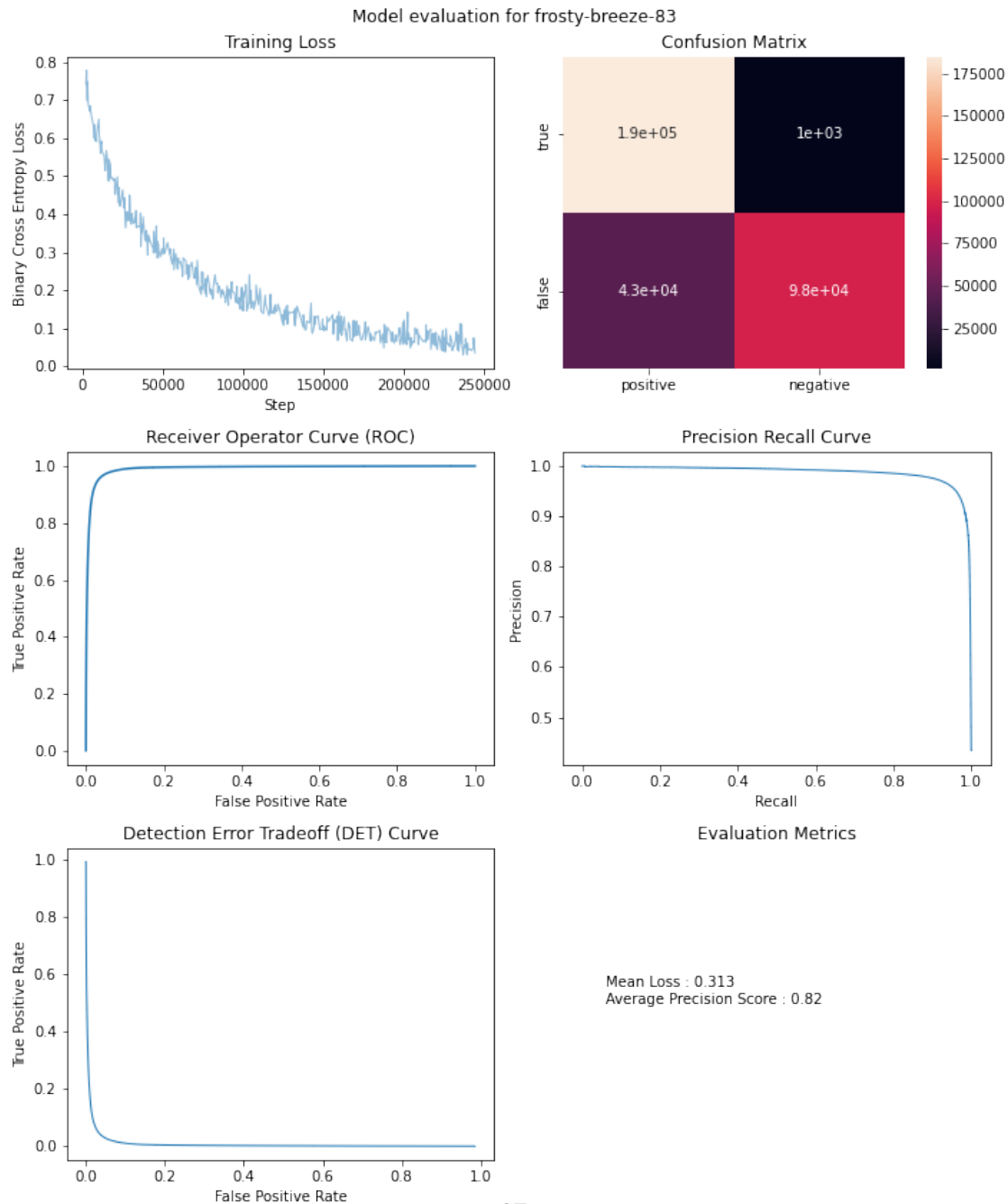
Figure 7:   Model evaluation for another instance of *frosty-breeze-81*, this time designated *devout-thunder-90*, again retrained on the lab-based screening dataset `screening-data.train.csv` and evaluated on `screening-data.test.csv`. The report uses the same methods as in figure 6
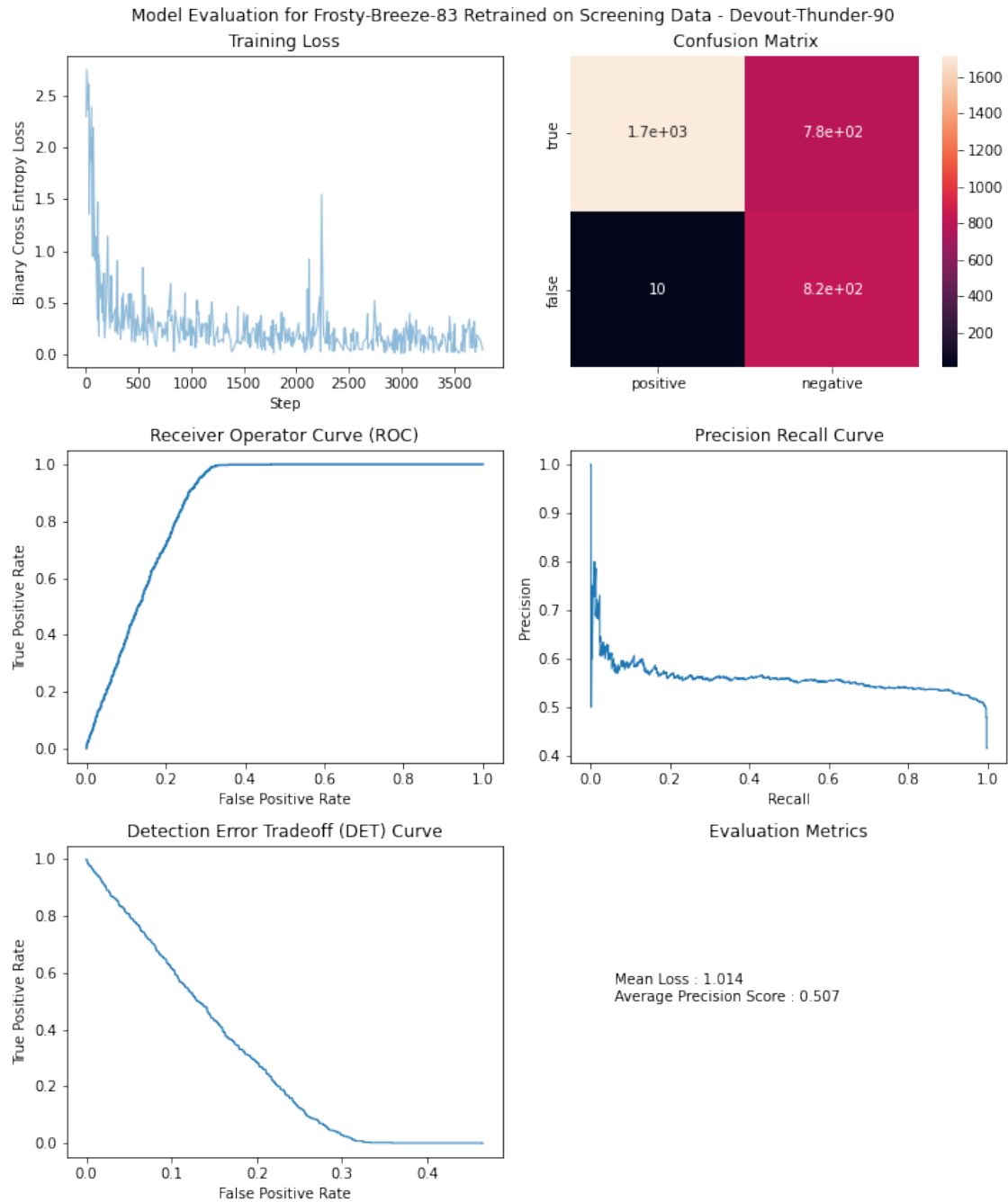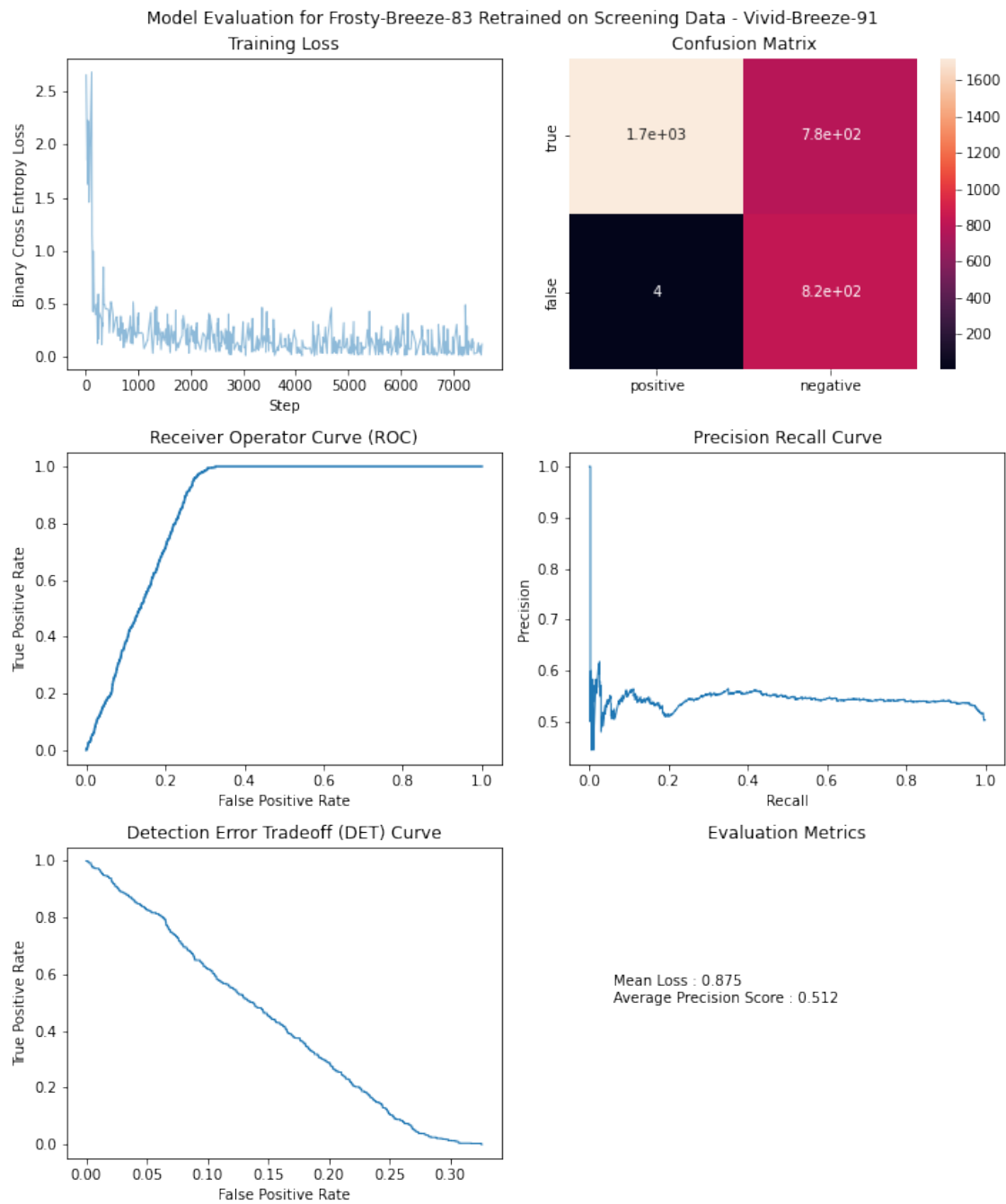
Figure 8: Model evaluation report after pre-training for the model designated *vivid-breeze-91*.

[3]   Christopher JC Whitehouse, Stephen G Bell, and Luet-Lok Wong. "P450 BM3 (CYP102A1): connecting the dots". In: *Chemical Society Reviews* 41.3 (2012), pp. 1218–1260.

[4]   Christopher F Butler et al. "Key mutations alter the cytochrome P450 BM3 conformational landscape and remove inherent substrate bias". In: *Journal of Biological Chemistry* 288.35 (2013), pp. 25387–25399.

[5]   Andrew Ng. "Nuts and bolts of building AI applications using Deep Learning". In: *NIPS Keynote Talk* (2016).

[6]   Roshan Rao et al. "Evaluating Protein Transfer Learning with TAPE". In: *bioarxiv* (2019).