Journal Title Here, 2022, pp. 1–34

doi: DOI HERE
Advance Access Publication Date: Day Month Year
Paper

PAPER

AutoPeptideML: A study on how to build more trustworthy peptide bioactivity predictors

Raúl Fernández-Díaz,[1,2,3,4,*] Rodrigo Cossio-Pérez,[2,3,5] Clement Agoni,[2,3,6] Hoang Thanh Lam,[1] Vanessa Lopez[1] and Denis C. Shields[2,3,*]

[1]IBM Research, Dublin, IBM Technology Campus Damastown Industrial Park Mulhuddart, D15 HN66, Dublin, Ireland, [2]School of Medicine, University College Dublin, Belfield, D04 C1P1, Dublin, Ireland, [3]Conway Institute of Biomolecular and Biomedical Science, University College Dublin, Belfield, D04 C1P1, Dublin, Ireland, [4]The SFI Centre for Research Training in Genomics Data Science, [5]Department of Science and Technology, National University of Quilmes, Roque Sáenz Peña 352, Bernal, B1876, Provincia de Buenos Aires, Argentina and [6]Discipline of Pharmaceutical Sciences, School of Health Sciences, University of KwaZulu-Natal, Durban, 4000, South Africa
*Corresponding authors. raul.fernandezdiaz@ucdconnect.ie, denis.shields@ucd.ie
FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

Motivation

Automated machine learning (AutoML) solutions can bridge the gap between new computational advances and their real-world applications by enabling experimental scientists to build their own custom models. We examine different steps in the development life-cycle of peptide bioactivity binary predictors and identify key steps where automation can not only result in a more accessible method, but also more robust and interpretable evaluation leading to more trustworthy models.

Results

We present a new automated method for drawing negative peptides that achieves better balance between specificity and generalisation than current alternatives. We study the effect of homology-based partitioning for generating the training and testing data subsets and demonstrate that model performance is overestimated when no such homology correction is used, which indicates that prior studies may have overestimated their performance when applied to new peptide sequences. We also conduct a systematic analysis of different protein language models as peptide representation methods and find that they can serve as better descriptors than a naive alternative, but that there is no significant difference across models with different sizes or algorithms. Finally, we demonstrate that an ensemble of optimised traditional machine learning algorithms can compete with more complex neural network models, while being more computationally efficient. We integrate these findings into AutoPeptideML, an easy-to-use AutoML tool to allow researchers without a computational background to build new predictive models for peptide bioactivity in a matter of minutes.

Availability and Implementation

Source code, documentation, and data are available at `https://github.com/IBM/AutoPeptideML` and a dedicated web-server at `http://peptide.ucd.ie/AutoPeptideML`.

Contact and Supplementary Information

raul.fernandezdiaz@ucdconnect.ie or denis.shields@ucd.ie. Supplementary Information can be accessed from Zenodo: `https://zenodo.org/records/13363975`

Introduction

Peptides are short amino acid chains with 3 to 50 residues with a great variety of therapeutical properties. They have gained a lot of attention from the pharmaceutical and food industries, as their versatility makes them excellent candidates for drug or nutraceutical discovery (Wang et al., 2022). In this context, there is a growing demand for predictive models that can accelerate the discovery or design of peptides targeting new properties or bioactivities (Attique et al., 2020).

Novel developments in machine learning algorithms have offered new models for predicting protein structure (Lin et al.,

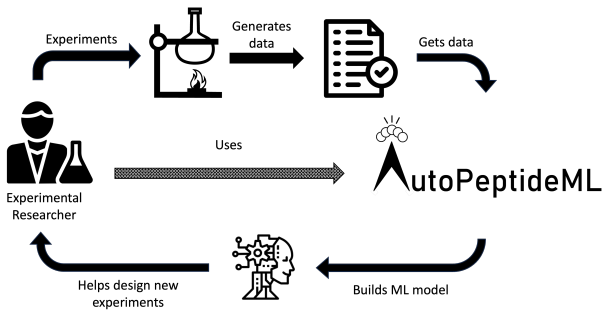

Fig. 1. Integration of AutoPeptideML in an experimental workflow. AutoPeptideML allows experimental researchers to build new custom models from their data. These models can then be used to propose new experiments. The data generated from these experiments can, in turn, be used to generate a new improved model. This leads to a feedback loop where the experimentation is guided towards more relevant peptides with each iteration.

2022) or different molecular properties (Dara et al., 2022). Despite these advancements, developing and evaluating new models is still an arduous process that requires both domain expertise and technical skills (Attique et al., 2020). Thus, most predictive models target broad and general applications, while solutions for more narrow use cases, like specific peptide bioactivities, remain underdeveloped (Attique et al., 2020).

Here, we propose building an automated machine learning (AutoML) system to automate the development of custom bioactivity predictors. There are several benefits that the introduction of such a system would provide. First, AutoML solutions can reduce the time required for model development from weeks to hours (He et al., 2021). Second, they help democratize machine learning by enabling researchers without a computational background to build effective models (He et al., 2021). Third, they help to ensure that best practices are followed, which can lead to increased trust in ML predictors within the field (Amirian et al., 2021). Fourth, they can greatly simplify and accelerate current strategies that require not only strong computational skills, but also very tedious experimentation through extensive trial-and-error (Attique et al., 2020). Overall, an AutoML tool for building peptide bioactivity predictors will allow experimental researchers to seamlessly introduce advanced modelling techniques into their experimental workflows in a matter of hours (see Figure 1).

A review of the existing literature revealed five key steps in the development of a peptide bioactivity predictor that could be benefited by this automation: 1) data gathering for negative peptides, 2) dataset partitioning, 3) computational representation of the peptides, 4) model training and hyperparameter optimisation, and 5) reporting of model evaluation.

1) Data gathering for the negative peptides. Binary classifiers require both positive and negative examples. Finding positive examples for peptide bioactivity is relatively simple as there is prior literature describing the function and role of a multitude of peptides (Quiroz et al., 2021). However, there are few repositories enumerating peptides that do not present a certain function or property (Attique et al., 2020). Further, there is no consensus in the literature as to how negative peptides should be chosen: some works opt for choosing fragments of proteins (Agrawal et al., 2021; Manavalan et al., 2019; Bin et al., 2020), others look for actual peptides (Agrawal et al., 2021; Pinacho-Castellanos et al., 2021; Pang et al., 2022; Charoenkwan et al., 2022c), and yet others use peptides with a known bioactivity that is different from their target (Agrawal et al., 2021; Charoenkwan et al., 2020a; Olsen et al., 2020).

If we consider the first and second approaches, the model learns to differentiate between peptides with the target bioactivity and random sequences (either protein fragments or peptides). However, the problem is that the model can exploit multiple confounding factors that do not have a direct bearing on the specific bioactivity, but that are related to the differences between generally bioactive peptides and random sequences. In the third approach, the opposite is true, positive and negative peptides may be so similar to each other that the model will be biased towards specific differential features between both bioactivities, hindering its ability to generalise.

In this paper, we explored introducing an intermediate solution: to draw the negative peptides from a database with multiple bioactivities. This approach generates a distribution of negative peptides that is as unbiased as possible (by covering several distinct bioactivities) as to generalise adequately, but that is similar enough to the positive peptide distribution (by also being bioactive peptides) as to minimise confounding factors.

2) Dataset partitioning. To evaluate predictive models it is necessary to divide the data into at least three distinct subsets: training, validation, and testing. The independence between the training and testing subsets is essential to obtain a reliable estimation of the future model performance (Walsh et al., 2021). To achieve this independence, community guidelines recommend building testing sets that do not share homologous sequences with the training set either by homology reduction (Walsh et al., 2021) or homology partitioning (Teufel et al., 2023; Fernández-Díaz et al., 2024). Despite this, most of the peptide bioactivity predictors reviewed (Agrawal et al., 2021; Bin et al., 2020; Pinacho-Castellanos et al., 2021; Charoenkwan et al., 2022c, 2020a, 2021; Xiao et al., 2021; Charoenkwan et al., 2022b; Rajput et al., 2015; Wei et al., 2020; Charoenkwan et al., 2020b) do not introduce any correction for homology when partitioning their datasets and those that do (Manavalan et al., 2019; Olsen et al., 2020; Zhang et al., 2022; Dai et al., 2021; Charoenkwan et al., 2022a; Chen et al., 2022), use high thresholds (80-90% of sequence identity, Table S1), still allowing for similar sequences to be in different sets, which could lead to the overestimation of model performance due to data leakage.

The main difference between homology-based reduction and homology-based partitioning is that in homology-based reduction a first clustering step is performed and only the centroids of the clusters are preserved. In partitioning algorithms, clusters are moved to different partitions so that the sequences within each partition do not have any neighbours in other partitions (e.g., no sequences in the testing set can have a neighbour in the training set). While previous methods (Teufel et al., 2023) removed a minimal number of sequences to ensure that partitions are completely independent. We rely on CCPart (Fernández-Díaz et al., 2024), a recently developed algorithm that is able to generate independent test sets without sequence removal. With this in mind, we explored the effects of introducing this homology-based dataset partitioning for building testing subsets more suited for evaluating model generalisation.

3) Computational representation of the peptides. For a predictive model to be able to interpret the peptide sequences, they need to be translated into mathematical objects (vectors

or matrices). The reviewed literature offers different options for performing this transformation that include statistics of: residue composition (Bin et al., 2020; Olsen et al., 2020; Charoenkwan et al., 2021), evolutionary profile (Wei et al., 2021), or physico-chemical properties (Bin et al., 2020; Pinacho-Castellanos et al., 2021; Charoenkwan et al., 2020b, 2022a). The consensus that can be drawn from the variety of different descriptor combinations is that each predictive task will require a different set of descriptors. Finding the optimal combination is a crucial and intricate step in the modelling process (Attique et al., 2020).

The advent of Protein Language Models (PLMs) like the ESM (evolutionary scale modelling) (Rao et al., 2020; Lin et al., 2023) or RostLab (ProtBERT, Prot-T5-XL, or ProstT5) (Elnaggar et al., 2021; Heinzinger et al., 2023) families has allowed for much simpler and richer protein representations. Given a sequence s, these models have learned the probability that a residue will appear in position i given the rest of the sequence $\{s - r_i\}$, $P(r_i|\{s - r_i\})$. This probability is related to the concept of conserved and unconserved positions that is often used when analysing multiple sequence alignments (Lin et al., 2022). The models are trained on a vast set of sequences from the UniRef (Suzek et al., 2007) or BFD (Steinegger et al., 2019) databases which include not only protein sequences, but also peptides. Moreover, at least, two prior studies have demonstrated that they can be used for representing peptides outperforming traditional description strategies (Du et al., 2023; Dee, 2022). However, there are many PLMs varied both in terms of size and learning method and it is not clear which may be the optimal choice for computing peptide representations. In this paper, we continue this line of research by addressing two questions: does model size have an impact on how suitable their representations are for describing peptides? and is there any significant difference between different classes of models?

4) Model training and hyperparameter optimisation. There are many different algorithms for fitting predictive models to a binary classification task and choosing between them is an extended trial-and-error task. Here, we considered an alternative approach: to use standard tools in the AutoML domain for performing hyperparameter bayesian optimisation of simple machine learning models and ensembling them.

5) Model evaluation reporting. The final step in the development of any predictive model is to report how reliable the future predictions of the method are going to be. The main goal of our automated process is to enable researchers without a computational background to leverage the tools, therefore, we structure the output of the pipeline to provide all necessary information for reproducing the training of the model and a summary that offers guidance in how to interpret the different evaluation scores of the model.

These contributions have been integrated into a computational tool and webserver, named AutoPeptideML, that allows any researcher to build their own custom models for any arbitrary peptide bioactivity they are interested in. The webserver requires only minutes to build a predictor and its use is as simple as uploading a dataset with positive examples. It provides an output summary that facilitates the interpretation of the reliability of the predictor generated and it has an additional window supporting the use of the generated models for predicting the bioactivity of any given set of peptides.

Materials and methods

Data acquisition. 18 different peptide bioactivity datasets containing positive and negative samples were used to evaluate the effect of the different methods. These datasets were selected from a previous study, considering the use of the ESM2-8M PLM for general peptide bioactivity prediction (Du et al., 2023). The datasets ranged in size from 200 to 20,000 peptides (see Table 1). Here, they are referred to as the "original" datasets.

Dataset with new negative peptides. For each of the original datasets, a new version was constructed using the new definition of negative peptides, termed "NegSearch". The negative peptides were drawn from a curated version of the Peptipedia database "APML-Peptipedia" comprised of 92,092 peptides representing 128 different activities (see Figure S1). To avoid introducing false negative peptides into the negative subset, all bioactivities that may overlap with the bioactivity of interest were excluded (see Table S1). To ensure that the negative peptides were drawn from a similar distribution to the positive peptides and thus minimise the number of confounding factors, for each dataset we calculated a histogram of the lengths of its peptides with bin size of 5. Then, for each bin in the histogram, we queried APML-Peptipedia for as many peptides as present in the bin, with lengths between its lower and upper bounds. If there were not enough peptides, the remaining peptides were drawn from the next bin.

Dataset partitioning. Two different partitioning strategies were used to generate the training/testing subsets: A) random partitioning and B) CCPart (Fernández-Díaz et al., 2024), a novel homology-based partitioning algorithm which creates an independent testing set ensuring that there are no homologous sequences between training and testing. Briefly, the algorithm calculates pairwise alignments among all dataset sequences to form a pairwise similarity matrix. It then clusters these sequences based on the similarity matrix using the connected components algorithm (Fernández-Díaz et al., 2024). Lastly, it iteratively transfers the smallest clusters to the testing set until it reaches the desired size (in our case, 20% of total sequences). This process ensures that there are no sequences in the testing set similar to those in the training set. The datasets generated through this strategy are referred to as "NegSearch+HP".

The CCPart algorithm (Fernández-Díaz et al., 2024) achieves two main objectives: 1) it creates a test set completely independent from training, insofar there are no sequences in the test set that are similar (as defined by the threshold) to those in the training set, and 2) it selects a test set that is as different from the training distribution as possible. This second objective is achieved by selecting the smallest clusters. The advantage of this decision is that because the clusters selected are small, we can fit more of them into the test set improving its diversity. On the other hand, the smaller a cluster is, the fewer neighbours it has and, consequently, the more unique the sequence is within the dataset. Overall, the algorithm attempts to simulate the real world scenario where the model is used to predict sequences different from those in the training set.

In both cases, A) randomly partitioned or B) homology partitioned, the training set is further subdivided into 10 folds for cross-validation. This second division relies on random stratified partitioning, to create 10 cross-validation folds.

Pairwise sequence alignments. The pairwise sequence alignments were calculated using the MMSeqs2 software with prior k-mer prefiltering (Steinegger and Söding, 2017). We

Table 1. Original benchmark datasets. SOTA Ref: Reference to best reported model in the literature.

Dataset	Negative Class	Partitioning	Number positives	SOTA Ref
Antibacterial (Pinacho-Castellanos et al., 2021)	Random peptides	Homology maximisation	8,278	(Pinacho-Castellanos et al., 2021)
ACE inhibitor (Manavalan et al., 2019)	Random protein fragments	Homology reduction (90%)	1,299	(Manavalan et al., 2019)
Anticancer 1 (Agrawal et al., 2021)	Antimicrobial peptides	Random	861	(Charoenkwan et al., 2021)
Anticancer 2 (Agrawal et al., 2021)	Random protein fragments	Random	970	(Agrawal et al., 2021)
Antifungal (Pinacho-Castellanos et al., 2021)	Random peptides	Homology maximisation	993	(Pinacho-Castellanos et al., 2021)
Antimalarial 1 (Xiao et al., 2021)	Random peptides	Random	139	(Charoenkwan et al., 2022c)
Antimalarial 2 (Agrawal et al., 2021)	Random protein fragments	Random	139	(Charoenkwan et al., 2022c)
Antimicrobial (Pinacho-Castellanos et al., 2021)	Random peptides	Homology maximisation	6,460	(Pinacho-Castellanos et al., 2021)
Antioxidant (Olsen et al., 2020)	Experimental + random peptides	Homology reduction (90%)	728	(Olsen et al., 2020)
Antiparasitic (Zhang et al., 2022)	Random peptides	Homology reduction (90% for positives and 60% for negatives)	301	(Zhang et al., 2022)
Antiviral (Pinacho-Castellanos et al., 2021)	Random peptides	Homology maximisation	2,944	(Pinacho-Castellanos et al., 2021)
Brain-blood barrier crossing (Dai et al., 2021)	Random peptides	Homology reduction (90%)	119	(Dai et al., 2021)
DPPIV inhibitors (Charoenkwan et al., 2020a)	Random + Bioactive	Random	665	(Charoenkwan et al., 2022b)
Anti-MRSA (Charoenkwan et al., 2022a)	Random peptides	Homology reduction (80%)	148	(Charoenkwan et al., 2022a)
Neuropeptide (Bin et al., 2020)	Random protein fragments	Homology reduction (90%)	2,425	(Chen et al., 2022)
Quorum sensing (Rajput et al., 2015)	Random peptides	Random	220	(Wei et al., 2020)
Toxic (Wei et al., 2021)	Random peptides	Random	1,932	(Wei et al., 2021)
Tumor T-cell antigens (Charoenkwan et al., 2020b)	T-cell antigens not associated to disease	Random	592	(Charoenkwan et al., 2020b)

considered that two peptides were similar if they had a sequence identity above 30% using the longest sequence as denominator.

Peptide representations. In order to evaluate the PLM peptide representations, the following methods (Rao et al., 2020; Lin et al., 2023; Elnaggar et al., 2021; Heinzinger et al., 2023) were evaluated: ESM2-8M, ESM2-35M, ESM2-150M, ESM2-650M, ESM1b, ProtBERT, Prot-T5-XL-UniRef50, ProstT5 (sequence mode), and one-hot encoding as a non-PLM-based baseline.

PLMs generate as output a matrix M with shape $n \times e$, where n is the number of residues in the peptide and e is the model embedding size (in this study $e \in [320, 1280]$, depending on the model). Each row in this matrix corresponds to a residue-level representation. We obtain a peptide-level representation r by averaging across all residues: $r = \frac{1}{n}\sum_{i=1}^{n} M_i$ (Du et al., 2023; Dee, 2022). Please note that r is a vector with e dimensions.

Model training and hyperparameter optimisation. In order to evaluate the model training and hyperparameter optimisation step, hyperparameter optimisation through bayesian optimisation (Akiba et al., 2019) was performed separately for K-nearest neighbours (KNN), light gradient boosting machine (LightGBM) and random forest classifier (RFC) and all models were ensembled (see Table S2 for more details about the hyperparameter optimisation). The optimisation aims to maximise model performance (measured as the average Matthew's correlation coefficient (Chicco et al., 2021) across the 10 cross-validation folds). The optimisation was conducted separately for each of the three models, leading to one optimal hyperparameter configuration per algorithm (three in total). After hyperparameter optimisation, each of the three models was trained against each of the 10 cross-validation folds using the optimal configuration. Thus, the final ensemble contained 10 instances (one per cross-validation fold) of each of the three models for a total of $3 \times 10 = 30$ models.

Final ensemble predictions were the average of all 30 individual predictions. This strategy is referred to as "Optimised ML Ensemble" or OMLE throughout the text. The three learning algorithms we used were chosen to provide a diverse representation of simple machine learning algorithms with computationally efficient implementations. We decided to use an ensemble, because it has been shown that for small datasets it leads to more robust predictors (Dvornik et al., 2019).

Our system was compared against an amended version of the UniDL4BioPep (Du et al., 2023) framework, which we named "UniDL4BioPep-A" (more details about the architecture of the model can be found in the original publication (Du et al., 2023)

and are summarised in Table S3). This amendment differs from the original in that, following community guidelines (Walsh et al., 2021), it used 10-fold cross-validation to determine the best possible checkpoint, instead of the hold-out testing set.

Every training experiment was run three times in order to get a crude estimation of the variability between experiment replications. The number of replicates is too small for proper statistical significance comparison, but the experimental design was constrained by the computational cost of each individual experiment run.

Model evaluation metrics. Model performance was measured in terms of Matthew's correlation coefficient (MCC), which is a binary classification metric that is specially recommended for measuring model performance in datasets with imbalanced labels (different number of positive and negative samples) (Chicco et al., 2021; Chicco and Jurman, 2023). Most datasets considered in the study have a balanced number of positive and negative labels. Therefore, for the purposes of model evaluation, we have defined any prediction with a probability score greater than 0.5 as positive and lower or equal as negative.

Calculation of peptide physico-chemical properties. To better describe the composition of the datasets used throughout the study we calculated the distribution of different physico-chemical properties of the peptides. The properties considered were: the aliphatic index using the method described by (Ikai, 1980); the Boman potential interaction index using the method described by (Boman, 2003); the charge and isolectric point using the methods described by (Sillero and Maldonado, 2006); the hydrophobic moment using the method described by (Eisenberg et al., 1984); and the predicted structural class using the method described by (Zhou and Assa-Munt, 2001). The calculations of all aforementioned algorithms were performed using the corresponding implementations available in (Larralde, 2024) with default settings.

Results and Discussion

We have focused our study of peptide bioactivity prediction in the binary classification task of discriminating between peptides that show a specific biologically-relevant property or function and those which do not. There are three reasons informing this choice: i) there are more datasets available for binary classification than regression, making the benchmarking more comprehensive, ii) the intepretation of regression metrics like root mean squared error (RMSE) is less intuitive than metrics for binary classification in balanced datasets, and therefore less suitable for the target non-expert audience, and iii) multi-class or multi-label problems can be formalised as sets of binary classification problems (García-Pedrajas and Ortiz-Boyer, 2011).

Effect of the sampling strategy for gathering negative peptides

We started by examining the effect of the new sampling strategy for gathering negative peptides.

The new negative peptides have a distribution of physico-chemical properties more similar to that of the positives. We have first examined the hypothesis that sampling negative peptides from the APML-Peptipedia database of bioactive peptides would lead to negative peptides with distributions more similar to those of the positive peptides. We calculated the distributions for different physico-chemical properties of the datasets and compared the positives with the original negatives and the new negatives (see Figures S2-S17).

The results show that, generally, the distribution of the new negatives is closer to the distribution of the positive peptides, specifically in the cases of the Antibacterial, ACE inhibitor, Antifungal, Antimalarial, Antimicrobial, Antioxidant, Antipara-sitic, Brain-blood barrier crossing, DPPIV inhibitor, and Toxicity. In the rest of the datasets, the distribution resembles much more closely that of the original negatives.

The introduction of the new negative peptides leads to more challenging modelling problems. We then examined the effect that the introduction of the new negative peptides had in the complexity of the modelling problem. With everything else being identical, Figure 2 clearly shows how the introduction of the new negative peptides leads to a more challenging modelling problem, this is most likely due to the reduction of confounding factors that the model can exploit to discriminate between the positive and negative classes.

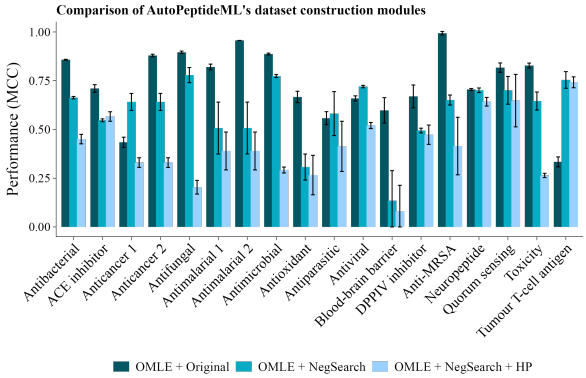

Fig. 2. Evaluation of AutoPeptideML's dataset construction modules. Error bars reflect the standard deviation across three replicates. OMLE: Optimised ML ensemble; Original: Original benchmark; NegSearch: Dataset with new negative peptides; HP: Homology-based dataset partitioning module.

It is particularly interesting to consider the change in the dataset pairs Anticancer 1 and 2, and Antimalarial 1 and 2 as the original datasets were built from the same sets of positive peptides, but relied on different sampling strategies for obtaining the negative peptides. Briefly, Anticancer 1 drew its negatives from a database of antimicrobial peptides (which are known to overlap with the anticancer peptides (Tornesello et al., 2020)) and Anticancer 2 drew them from a database of random protein fragments. This is reflected in Figure 2 where Anticancer 1 - Original demonstrates lower performance than Anticancer 2 - Original. It is noteworthy that the NegSearch dataset achieves an intermediate performance between them, as was to be expected from the NegSearch negative sampling being an intermediate definition between a specific bioactivity and random peptide sequences. Similarly, Antimalarial 1 - Original draws its negative peptides from a collection of random peptides, while Antimalarial 2 - Original draws them from a collection of protein fragments. Figure 2 shows how the more restrictive that the negative peptide sampling is the lower the model apparent performance, with Antimalarial 2 - Original achieving the highest apparent performance and Antimalarial - NegSearch the lowest.

Overall, the results indicate that the choice of sampling method for acquiring the negative peptides has an important effect on the perceived model performance. In the end, the optimal sampling method will depend in the intended use for the model and whether it will be applied to protein fragments, random peptides or to distinguish between different peptide bioactivities. However, our experiments suggest that sampling from a collection with peptides with diverse bioactivities offers a balance between specificity and future generalisation, particularly, when the future target distribution is unknown at time of model development.

Effect of homology-based partitioning.

We next examined the effect of introducing the homology-based partitioning algorithm for generating the training and test sets.

The training and test sets are not independent in the original datasets. We started by analysing the interdependence between training and test sets in the original datasets. We measured interdependence as the proportion of peptides in the training set with at least one similar peptide in the test set. We classified two peptides as similar if they have more than 30% sequence identity in the pairwise local alignment. The results compiled in Table 2 indicate that for 13 of the 18 original datasets, at least 10% of the peptides in the training set are similar to sequences in the testing set, compromising their independence. If we consider the datasets (see Table 1) for which homology-based correction was used (ACE inhibitor, Antioxidant, Antiparasitic, Anti-MRSA, and Neuropeptide), we observe that only two of them have less than 10% interdependence, which highlights the need for introducing similarity correction techniques at low thresholds.

The introduction of the new negatives does not reduce the interdependence between training and test sets, but it can even increase in some cases. The biggest increments are observed in the Antibacterial, Antifungal, Antimicrobial, Antiviral, and Antiviral datasets. These datasets all have been partitioned using a homology maximisation algorithm (see Table 1) that creates a test set with representatives from all clusters in training, this representatives are evenly sampled and thus the interdependence between the two subsets while not reduced it is bounded. In the "NS" datasets, training and test sets are randomly partitioned, and therefore members from highly connected clusters can be overrepresented in the test set, thus leading to the observed increase in the interdependence.

Independent test sets lead to more challenging evaluation. Table 2 shows how that the training and test sets are completely independent from each other when the CCPart algorithm is used for homology-based dataset partitioning.

Figure 2 clearly shows how the independent test sets lead to a much more challenging evaluation with a significant drop in model performance in most datasets. This result suggests that previous studies have tended to overestimate the performance of the models when applied to real-world sequences different from those present in their training set.

In this study, we have considered as similarity metric the sequence identity in local pairwise alignments when performing the homology-based partitioning, following the prior studies that introduced any type of homology-based correction technique referenced in Table 1, which all use sequence identity. The results obtained showcase the importance of accounting for the similarity between training and test partitions to properly evaluate model out-of-distribution generalisation. Peptides being entities halfway between proteins and small molecules, also support other similarity metrics based on their physico-chemical properties or chemical structure that may be more suitable for partitioning (Fernández-Díaz et al., 2024; Orsi and Reymond, 2024). We keep the exploration of alternative similarity metrics for peptide dataset partitioning for future work.

Table 2. Training-testing interdependence analysis of all datasets. The percentages correspond to the proportion of training sequences with at least one similar sequence (sequence identity $> 30\%$) in the testing set. Columns correspond to the different datasets constructed: Original datasets, the NegSearch datasets (NS) and the NegSearch datasets with homology-based partitioning (NS+HP). *: Equivalent to Anticancer 1 for NS and NS+HP; **: Equivalent to Antimalarial 1 for NS and NS+HP

Dataset	Original	NS	NS+HP
Antibacterial	36%	64%	0%
ACE inhibitor	1%	3%	0%
Anticancer 1	59%	50%	0%
Anticancer 2*	30%	∼	∼
Antifungal	34%	58%	0%
Antimalarial 1	41%	15%	0%
Antimalarial 2**	10%	∼	∼
Antimicrobial	47%	63%	0%
Antioxidant	1%	0%	0%
Antiparasitic	11%	32%	0%
Antiviral	26%	44%	0%
Blood-brain barrier	4%	4%	0%
DPPIV inhibitor	5%	4%	0%
Anti-MRSA	15%	34%	0%
Neuropeptide	24%	24%	0%
Quorum sensing	18%	9%	0%
Toxicity	56%	56%	0%
Tumor T-cell antigens	0%	3%	0%

Homology-based partitioning generates sufficiently diverse test sets. One possible problem with homology-based partitioning is that it may lead to the creation of test sets with very few similarity clusters, thus compromising evaluation. A comparison of the number of clusters present in the training and test set before and after the introduction of new negatives and homology-based partitioning (see Table S4) shows that though the number of clusters in the test set always diminishes, in most cases the number of clusters in the test set still represents approximately 20% of the number of clusters in training, and it never represents less than 10%. This confirms that the CCPart algorithm is able to generate sufficiently diverse test sets.

Protein Language Models as peptide representation methods

Recent studies have reported the use of PLMs for predicting peptide bioactivity (Du et al., 2023; Dee, 2022), however, they have not been compared to a naive baseline representation like one-hot encoding; nor has there been an evaluation on which PLM may be more suited for peptide representation. All experiments are conducted with the NegSearch+HP datasets to ensure that we were properly evaluating model generalisation.

Baseline. First, we compare the PLMs to a naive baseline representation (one-hot encoding). Figure 3 shows that generally PLMs are significantly better representation methods across datasets, though in specific cases one-hot encoding appears to achieve similar performance. There are different idiosyncrasies within those datasets that may explain the behaviour, for example in the Blood-brain barrier experiments

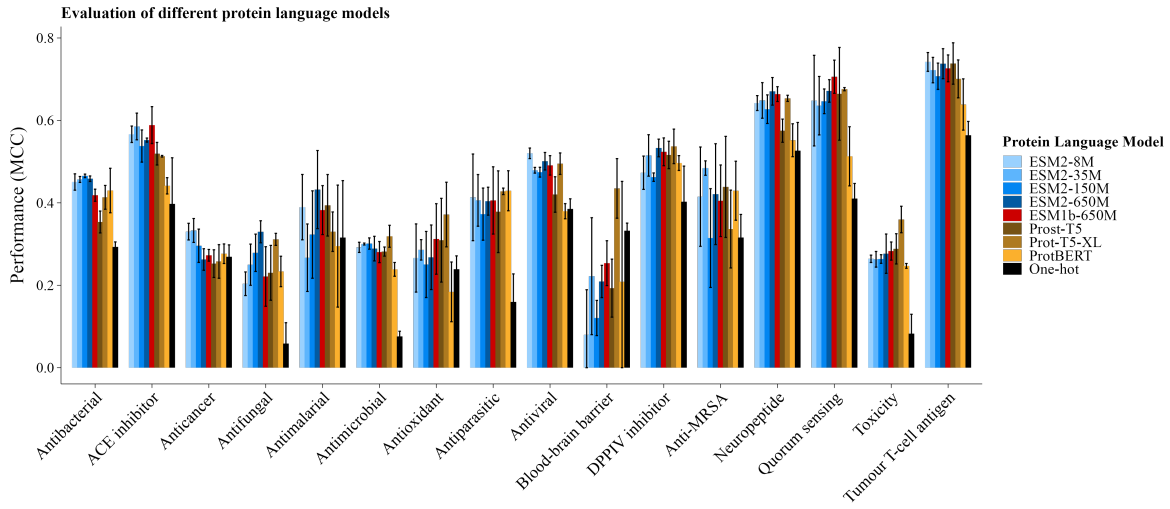

Fig. 3. Evaluation of different protein language models. Error bars reflect the standard deviation across three replicates.

the number of training peptides is really small (∼200, see Table 1) which leads to a lot of instability between the different runs (as can be seen by the size of the error bars).

Model size. We evaluated four different PLM models from the ESM family with increasing size: ESM2-8M (8 million parameters), ESM2-35M (35 million parameters), ESM2-150M (150 million parameters), ESM2-650M (650 million parameters). We also evaluated ESM1b-650M (650 million parameters), from a previous version of ESM. Figure 3 shows that there is no significant difference between models across all datasets and no correlation between model size and performance.

This observation appears contrary to the established consensus that bigger PLMs tend to perform better (Lin et al., 2023; Elnaggar et al., 2021; Rao et al., 2019). It is important to note that those studies focused on a very particular use of the PLMs known as full-model transfer learning (Li et al., 2024). However, in our experiments, we relied on representation transfer instead (Li et al., 2024; Unsal et al., 2022). The main difference between both regimes is that in full-model transfer learning every parameter in the model is adjusted (fine-tuned) for the downstream task, whereas in representation transfer, the internal parameters of the model do not change. There are two main consequences that derive from this distinction.

First, full-model fine-tuning tuning requires the model to run several times through the training data to iteratively optimise its internal parameters. This is a computationally intensive operation, and the cost increases with model size. Representation transfer, in contrast, only requires a single run through the training data to compute the representations and is thus much faster and does not require specialised hardware like GPUs. Furthermore, when working with small datasets sizes (like most peptide datasets), there is less risk of overfitting with representation transfer (only the parameters of the downstream model are optimised) than with full-model fine-tuning (where both the model parameters, $8 - 650 \times 10^6$, and the downstream models are optimised). We decided to focus on representation transfer for this study because of these two reasons.

Second, the more parameters a model has, the greater its learning capacity will be. This learning capacity can only be accessed in full-model fine-tuning as it allows for the optimisation of the internal parameters of the model. Thus, observing no correlation between model size and downstream performance in a representation transfer setting is not necessarily inconsistent with the prior literature. The question of whether the established PLM scaling rules for full-model transfer learning when modelling peptide sequences remains unanswered and is left for future work.

Type of model. We further compared the ESM models to the main models from the RostLab family: ProtBERT, Prot-T5-XL-UniRef50, and Prost-T5. Figure 3 shows that even though for certain datasets there might be significantly better models, when the effect is analysed across all datasets there is no significant difference between the different models or families.

All things considered, the ESM2-8M model achieves a commendable balance between enhanced performance relative to one-hot encoding and minimal computational requirements. In any case, the optimal performance will likely be achieved when traditional representation methods are combined with PLM representations. A systematic study on the optimal way to combine these types of features is kept for future work.

Optimised Machine Learning Ensemble and PLM representations as an alternative to highly engineered approaches

We compared the performance of two general purpose frameworks: a deep learning based model (UniDL4BioPep-A (Du et al., 2023) and Table S3) and our optimised machine learning ensemble (OMLE).

Comparison to handcrafted models. Figure 4.A (see Table S8 for alternative metrics) shows that when applied to a literature derived benchmark set of datasets, the two general purpose PLM-enabled bioactivity predictors have a performance comparable with the self-reported performance of the best handcrafted models for each specific dataset (see Table 1 for the reference of each of the models). Moreover, our proposed AutoML solution (OMLE) was able to out-perform the handcrafted models on 6 out of 17 benchmark datasets for which data was available and was not significantly different for another 2.

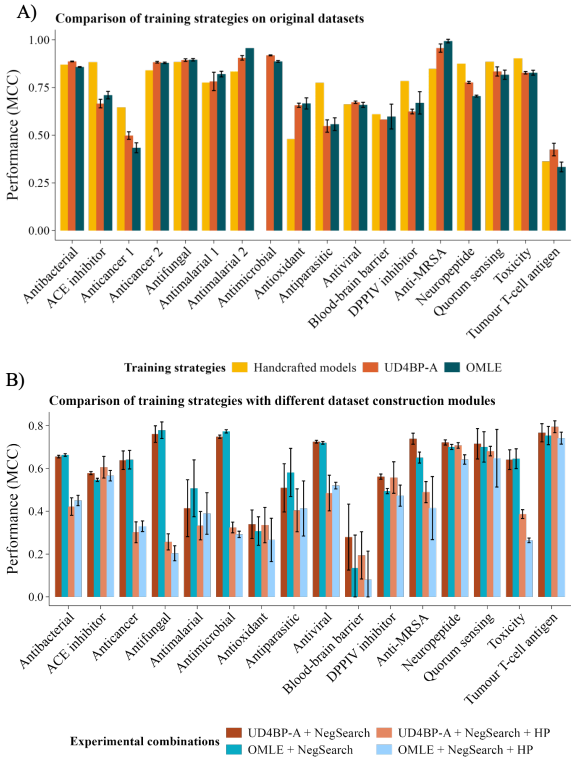

Fig. 4. A: **Comparison of training strategies on original datasets**. B: **Comparison of training strategies with different dataset construction modules**. Error bars reflect the standard deviation across three replicates. OMLE: Optimised ML ensemble; UD4BP-A: UniDL4BioPep-A

These results show that the combination of the PLM representations and the OMLE learning strategy, provide a fast, convenient, and competitive alternative to highly engineered approaches with handcrafted models and peptide representations that require both domain and technical expertise.

In any case, the results obtained with our approach, though solid with the out-of-the-box configuration, can be further improved by combining the PLM representations with traditional representations and defining OMLEs with wider or narrower sets of learning algorithms and hyperparameter spaces.

Comparison of an optimised ML ensemble with a neural network. When compared with Fig 4.A, Figure 4.B (see Table S9) shows that when the new sampling strategy for gathering negative peptides is introduced, both ML and neural network general purpose models show an equivalent drop in apparent performance, reflecting the more challenging task of predicting a specific bioactivity against peptides from a diverse collection of bioactivities. The performance drops further in both models when homology-based partitioning is introduced. Remarkably, there is no significant evidence of greater overfitting on the part of the DL model, despite the small dataset sizes, this might be due to the relatively small size of UniDL4BioPep-A. Overall, these results allow us to conclude that OMLE achieves comparable performance to a more complex neural network model, while being both more user-friendly and computationally efficient.

AutoPeptideML

All the findings described thus far, were used to guide the development of AutoPeptideML, a computational tool and webserver that allows researchers to easily build strong peptide bioactivity predictors and provide a robust evaluation that complies with community guidelines. Figure 5 provides an overview of the final AutoPeptideML workflow.

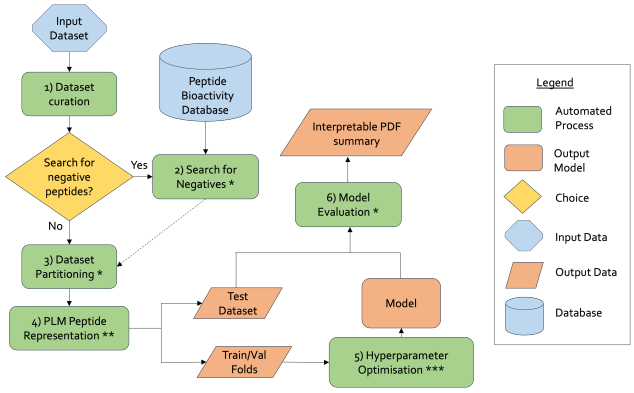

Fig. 5. Visual summary of the AutoPeptideML workflow. *: Steps based on the results showcased in Figure 2. **: Step based on the results showcased in Figure 3. ***: Step based on the results showcased in Figure 4.

The primary objective behind the design of AutoPeptideML is to provide a user-friendly tool that does not require extensive technical knowledge to use, while still remaining highly versatile. This is achieved through a pipeline that guarantees compliance with community guidelines such as DOME (Data, Optimisation, Model, and Evaluation) (Walsh et al., 2021), ensuring a robust scientific validation (see Supplementary G).

Users are free to define the number of models that should be included in the hyperparameter optimization, as well as their hyperparameter search space. AutoPeptideML supports the following algorithms: K-nearest neighbours (KNN), light gradient boosting (LightGBM), support vector machine (SVM), random forest classification (RFC), extreme gradient boosting (XGBoost), simple neural networks like the multi-layer perceptron (MLP), and 1D-convolutional neural networks (1D-CNN). Model selection and HPO are conducted simultaneously in a cross-validation regime so that the metric to optimise is the average across n folds. Thus, the system is never exposed to the testing set, which is kept unseen until the final model evaluation (Walsh et al., 2021). The system also supports all PLM models used throughout the study.

AutoPeptideML can be used in two regimes: *Model builder* and *Prediction*. In the first mode new predictive models are created automatically from a single file with known positive peptides for the bioactivity of interest. In the second mode, any predictive model generated through the *model builder* can then be used to predict for each peptide in a dataset the likelihood that it possesses the desired bioactivity.

The outputs that the program generates are:

- *Model builder:* When used to develop new predictors, AutoPeptideML outputs a model fitted to predict the bioactivity of interest, a folder with all information necessary for reproducing the model, and an interpretable summary of the model capabilities see Supplementary G.

- *Prediction:* AutoPeptideML can also be used to leverage existing predictors. In this case, it outputs a list of the problem peptides sorted in descending order of predicted bioactivity (higher bioactivity first) and a measure of the uncertainty of each prediction.

AutoPeptideML is a tool that enables teams of experimental researchers without access to modelling expertise to quickly and easily build and interpret custom models to integrate into their experimental workflows. It can also be helpful for computational researchers to generate quick and robust baselines at the early stages of a new modelling project against which to compare any new methods. Moreover, the AutoPeptideML Python API allows for using any combination of representations that the researcher may desire to include (both traditional and PLM-based). Thus, it can assist both domain experts and computational scientists by providing a flexible and easy-to-use end-to-end modelling pipeline, allowing the former to easily construct new models and the latter to quickly run experiments and compare between different representation methods or modelling algorithms within a robust and reproducible environment.

Conclusions

The definition of the negative class used for building peptide bioactivity predictors has a significant impact on the model performance of up to 40% and has to be controlled in order to properly interpret model predictions. Here we introduced a negative sampling strategy that gathers negative peptides from a collection of peptides with diverse bioactivities which has been shown to achieve a balance between the strengths and weaknesses of current methods in terms of the specificity and interpretability of the predictions, and the reliability of the estimation of future model performance.

The partitioning strategy used to generate training and test subsets impacts the evaluation of model generalisation significantly and the introduction of homology-based partitioning algorithms can lead to a drop in perceived model performance of up to 50% when compared to random partitioning. The magnitude of these effects suggests that the model performance has been overestimated in most previous studies.

Using protein language models (PLMs) for computing peptide representations is a significantly better strategy than using a one-hot encoding (a naive representation) for most of the datasets considered. This underscores the potential of PLMs to compute peptide representations, in line with previous studies. Surprisingly, there is no significant correlation between model size and the performance observed, nor among different models. This marks a first step towards understanding the limitations of PLM scaling rules as it pertains their use for modelling peptide sequences.

The combination of PLM peptide representations and an optimised ensemble of simple ML models reaches state-of-the-art performance when compared both to an alternative general-purpose-framework and dataset-specific, handcrafted models across a set of 18 different datasets. Furthermore, there is no significant difference between using an ensemble of simple ML algorithms and more complex DL algorithms (UniDL4BioPep-A), even though the former is more computationally efficient.

We present AutoPeptideML as a computational tool and webserver that allows researchers without technical expertise to develop predictive models for any custom peptide bioactivity. It also facilitates compliance with community guidelines for predictive modelling in the life-sciences. It is able to handle several key steps in the peptide bioactivity predictor development life-cycle including: 1) data gathering, 2) homology-based dataset partitioning, 3) model selection and hyperparameter optimisation, 4) robust evaluation, and 5) prediction of new samples. Further, the output is generated in the form of a PDF summary easily interpretable by researchers not specialised in ML; alongside a directory that ensures reproducibility by containing all necessary information for re-using and re-training the models. All data and code are made available to enable the reproducibility of the results in this work.

The foundational principles underlying the issues described and solutions implemented throughout this study are relevant for the application of trustworthy ML predictors for any other biosequence (e.g., DNA, RNA, proteins, peptides, DNA methylation, etc.) and their automation facilitates the rigorous evaluation and development of new models by researchers not specialised in ML.

Funding

RFD was supported by Science Foundation Ireland through the SFI Centre for Research Training in Genomics Data Science under Grant number 18/CRT/6214. RCP received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 778247 (IDPFun). CA was supported by Enterprise Ireland and received funding from the European Union Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 847402.

Acknowledgements

The authors thank Silvia González López for designing the AutoPeptideML logo, Marcos Martínez Galindo for his technical advice on how to to deploy the AutoPeptideML webserver, and Patrick Timmons for his insightful comments on an early version of the manuscript.

Competing interests

No competing interest is declared.

References

Agrawal, P., Bhagat, D., Mahalwal, M., Sharma, N., and Raghava, G. P. (2021). Anticp 2.0: an updated model for predicting anticancer peptides. *Briefings in bioinformatics*, 22(3):bbaa153.

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631.

Amirian, M., Tuggener, L., Chavarriaga, R., Satyawan, Y. P., Schilling, F.-P., Schwenker, F., and Stadelmann, T. (2021). Two to trust: Automl for safe modelling and interpretable deep learning for robustness. In *Trustworthy AI-Integrating Learning, Optimization and Reasoning: First International Workshop, TAILOR 2020, Virtual Event, September 4–5, 2020, Revised Selected Papers 1*, pages 268–275. Springer.

Attique, M., Farooq, M. S., Khelifi, A., and Abid, A. (2020). Prediction of therapeutic peptides using machine learning: computational models, datasets, and feature encodings. *IEEE Access*, 8:148570–148594.

Bin, Y., Zhang, W., Tang, W., Dai, R., Li, M., Zhu, Q., and Xia, J. (2020). Prediction of neuropeptides from sequence information using ensemble classifier and hybrid features. *Journal of proteome research*, 19(9):3732–3740.

Boman, H. G. (2003). Antibacterial peptides: basic facts and emerging concepts. *Journal of internal medicine*, 254(3):197–215.

Charoenkwan, P., Chiangjong, W., Lee, V. S., Nantasenamat, C., Hasan, M. M., and Shoombuatong, W. (2021). Improved prediction and characterization of anticancer activities of peptides using a novel flexible scoring card method. *Scientific reports*, 11(1):3017.

Charoenkwan, P., Kanthawong, S., Nantasenamat, C., Hasan, M. M., and Shoombuatong, W. (2020a). idppiv-scm: a sequence-based predictor for identifying and analyzing dipeptidyl peptidase iv (dpp-iv) inhibitory peptides using a scoring card method. *Journal of proteome research*, 19(10):4125–4136.

Charoenkwan, P., Kanthawong, S., Schaduangrat, N., Li', P., Moni, M. A., and Shoombuatong, W. (2022a). Scmrsa: a new approach for identifying and analyzing anti-mrsa peptides using estimated propensity scores of dipeptides. *ACS omega*, 7(36):32653–32664.

Charoenkwan, P., Nantasenamat, C., Hasan, M. M., Moni, M. A., Manavalan, B., Shoombuatong, W., et al. (2022b). Stackdppiv: A novel computational approach for accurate prediction of dipeptidyl peptidase iv (dpp-iv) inhibitory peptides. *Methods*, 204:189–198.

Charoenkwan, P., Nantasenamat, C., Hasan, M. M., and Shoombuatong, W. (2020b). ittca-hybrid: Improved and robust identification of tumor t cell antigens by utilizing hybrid feature representation. *Analytical biochemistry*, 599:113747.

Charoenkwan, P., Schaduangrat, N., Lio, P., Moni, M. A., Chumnanpuen, P., and Shoombuatong, W. (2022c). iamap-scm: A novel computational tool for large-scale identification of antimalarial peptides using estimated propensity scores of dipeptides. *ACS omega*, 7(45):41082–41095.

Chen, S., Li, Q., Zhao, J., Bin, Y., and Zheng, C. (2022). Neuropred-clq: incorporating deep temporal convolutional networks and multi-head attention mechanism to predict neuropeptides. *Briefings in Bioinformatics*, 23(5).

Chicco, D. and Jurman, G. (2023). The matthews correlation coefficient (mcc) should replace the roc auc as the standard metric for assessing binary classification. *BioData Mining*, 16(1):4.

Chicco, D., Tötsch, N., and Jurman, G. (2021). The matthews correlation coefficient (mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. *BioData mining*, 14(1):1–22.

Dai, R., Zhang, W., Tang, W., Wynendaele, E., Zhu, Q., Bin, Y., De Spiegeleer, B., and Xia, J. (2021). Bbppred: sequence-based prediction of blood-brain barrier peptides with feature representation learning and logistic regression. *Journal of Chemical Information and Modeling*, 61(1):525–534.

Dara, S., Dhamercherla, S., Jadav, S. S., Babu, C. M., and Ahsan, M. J. (2022). Machine learning in drug discovery: a review. *Artificial Intelligence Review*, 55(3):1947–1999.

Dee, W. (2022). Lmpred: Predicting antimicrobial peptides using pre-trained language models and deep learning. *Bioinformatics Advances*, 2(1):vbac021.

Du, Z., Ding, X., Xu, Y., and Li, Y. (2023). Unidl4biopep: a universal deep learning architecture for binary classification in peptide bioactivity. *Briefings in Bioinformatics*, page bbad135.

Dvornik, N., Schmid, C., and Mairal, J. (2019). Diversity with cooperation: Ensemble methods for few-shot classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3723–3731.

Eisenberg, D., Weiss, R. M., and Terwilliger, T. C. (1984). The hydrophobic moment detects periodicity in protein hydrophobicity. *Proceedings of the National Academy of Sciences*, 81(1):140–144.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., et al. (2021). Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127.

Fernández-Díaz, R., Hoang, T. L., Lopez, V., and Shields, D. C. (2024). Effect of dataset partitioning strategies for evaluating out-of-distribution generalisation for predictive models in biochemistry. *bioRxiv*, pages 2024–03.

García-Pedrajas, N. and Ortiz-Boyer, D. (2011). An empirical study of binary classifier fusion methods for multiclass classification. *Information Fusion*, 12(2):111–130.

He, X., Zhao, K., and Chu, X. (2021). Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622.

Heinzinger, M., Weissenow, K., Sanchez, J. G., Henkel, A., Steinegger, M., and Rost, B. (2023). Prostt5: Bilingual language model for protein sequence and structure. *bioRxiv*, pages 2023–07.

Ikai, A. (1980). Thermostability and aliphatic index of globular proteins. *The Journal of Biochemistry*, 88(6):1895–1898.

Larralde, M. (2024). althonos/peptides.py.

Li, F.-Z., Amini, A. P., Yue, Y., Yang, K. K., and Lu, A. X. (2024). Feature reuse and scaling: Understanding transfer learning with protein language models. *bioRxiv*, pages 2024–02.

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

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

Manavalan, B., Basith, S., Shin, T. H., Wei, L., and Lee, G. (2019). mahtpred: a sequence-based meta-predictor for improving the prediction of anti-hypertensive peptides using effective feature representation. *Bioinformatics*, 35(16):2757–2765.

Olsen, T. H., Yesiltas, B., Marin, F. I., Pertseva, M., García-Moreno, P. J., Gregersen, S., Overgaard, M. T., Jacobsen, C., Lund, O., Hansen, E. B., et al. (2020). Anoxpepred: using deep learning for the prediction of antioxidative properties of peptides. *Scientific Reports*, 10(1):21471.

Orsi, M. and Reymond, J.-L. (2024). One chiral fingerprint to find them all. *Journal of cheminformatics*, 16(1):53.

Pang, Y., Yao, L., Xu, J., Wang, Z., and Lee, T.-Y. (2022). Integrating transformer and imbalanced multi-label learning to identify antimicrobial peptides and their functional activities. *Bioinformatics*, 38(24):5368–5374.

Pinacho-Castellanos, S. A., García-Jacas, C. R., Gilson, M. K., and Brizuela, C. A. (2021). Alignment-free antimicrobial peptide predictors: improving performance by a thorough analysis of the largest available data set. *Journal of Chemical Information and Modeling*, 61(6):3141–3157.

Quiroz, C., Saavedra, Y. B., Armijo-Galdames, B., Amado-Hinojosa, J., Olivera-Nappa, Á., Sanchez-Daza, A., and Medina-Ortiz, D. (2021). Peptipedia: a user-friendly web application and a comprehensive database for peptide research supported by machine learning approach. *Database*, 2021.

Rajput, A., Gupta, A. K., and Kumar, M. (2015). Prediction and analysis of quorum sensing peptides based on sequence features. *PLoS One*, 10(3):e0120066.

Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, P., Canny, J., Abbeel, P., and Song, Y. (2019). Evaluating protein transfer learning with tape. *Advances in neural information processing systems*, 32.

Rao, R. M., Meier, J., Sercu, T., Ovchinnikov, S., and Rives, A. (2020). Transformer protein language models are unsupervised structure learners. *bioRxiv*.

Sillero, A. and Maldonado, A. (2006). Isoelectric point determination of proteins and other macromolecules: oscillating method. *Computers in biology and medicine*, 36(2):157–166.

Steinegger, M., Mirdita, M., and Söding, J. (2019). Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nature methods*, 16(7):603–606.

Steinegger, M. and Söding, J. (2017). Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028.

Suzek, B. E., Huang, H., McGarvey, P., Mazumder, R., and Wu, C. H. (2007). Uniref: comprehensive and non-redundant uniprot reference clusters. *Bioinformatics*, 23(10):1282–1288.

Teufel, F., Gíslason, M. H., Almagro Armenteros, J. J., Johansen, A. R., Winther, O., and Nielsen, H. (2023). Graphpart: homology partitioning for biological sequence analysis. *NAR genomics and bioinformatics*, 5(4):lqad088.

Tornesello, A. L., Borrelli, A., Buonaguro, L., Buonaguro, F. M., and Tornesello, M. L. (2020). Antimicrobial peptides as anticancer agents: functional properties and biological activities. *Molecules*, 25(12):2850.

Unsal, S., Atas, H., Albayrak, M., Turhan, K., Acar, A. C., and Doğan, T. (2022). Learning functional properties of proteins with language models. *Nature Machine Intelligence*, 4(3):227–245.

Walsh, I., Fishman, D., Garcia-Gasulla, D., Titma, T., Pollastri, G., Harrow, J., Psomopoulos, F. E., and Tosatto, S. C. (2021). Dome: recommendations for supervised machine learning validation in biology. *Nature methods*, 18(10):1122–1127.

Wang, L., Wang, N., Zhang, W., Cheng, X., Yan, Z., Shao, G., Wang, X., Wang, R., and Fu, C. (2022). Therapeutic peptides: Current applications and future directions. *Signal Transduction and Targeted Therapy*, 7(1):48.

Wei, L., Hu, J., Li, F., Song, J., Su, R., and Zou, Q. (2020). Comparative analysis and prediction of quorum-sensing peptides using feature representation learning and machine learning algorithms. *Briefings in Bioinformatics*, 21(1):106–119.

Wei, L., Ye, X., Xue, Y., Sakurai, T., and Wei, L. (2021). Atse: a peptide toxicity predictor by exploiting structural and evolutionary information based on graph neural network and attention mechanism. *Briefings in Bioinformatics*, 22(5):bbab041.

Xiao, X., Shao, Y.-T., Cheng, X., and Stamatovic, B. (2021). iamp-ca2l: a new cnn-bilstm-svm classifier based on cellular automata image for identifying antimicrobial peptides and their functional types. *Briefings in bioinformatics*, 22(6):bbab209.

Zhang, W., Xia, E., Dai, R., Tang, W., Bin, Y., and Xia, J. (2022). Predapp: predicting anti-parasitic peptides with undersampling and ensemble approaches. *Interdisciplinary Sciences: Computational Life Sciences*, pages 1–11.

Zhou, G. and Assa-Munt, N. (2001). Some insights into protein structural class prediction. *Proteins: Structure, Function, and Bioinformatics*, 44(1):57–59.

A. APML-Peptipedia

The original Peptipedia database integrates information from 30 peptide bioactivity databases collecting almost 97,331 bioactive peptides labelled with 128 bioactivities (version 29_03_2023). APML-Peptipedia is the result of removing all sequences with non-standard residues or without any known bioactivity and contains 92,092 peptides (see Supplementary). Figure 1 describes the distribution of the physicochemical properties of the peptides comprising APML-Peptipedia.

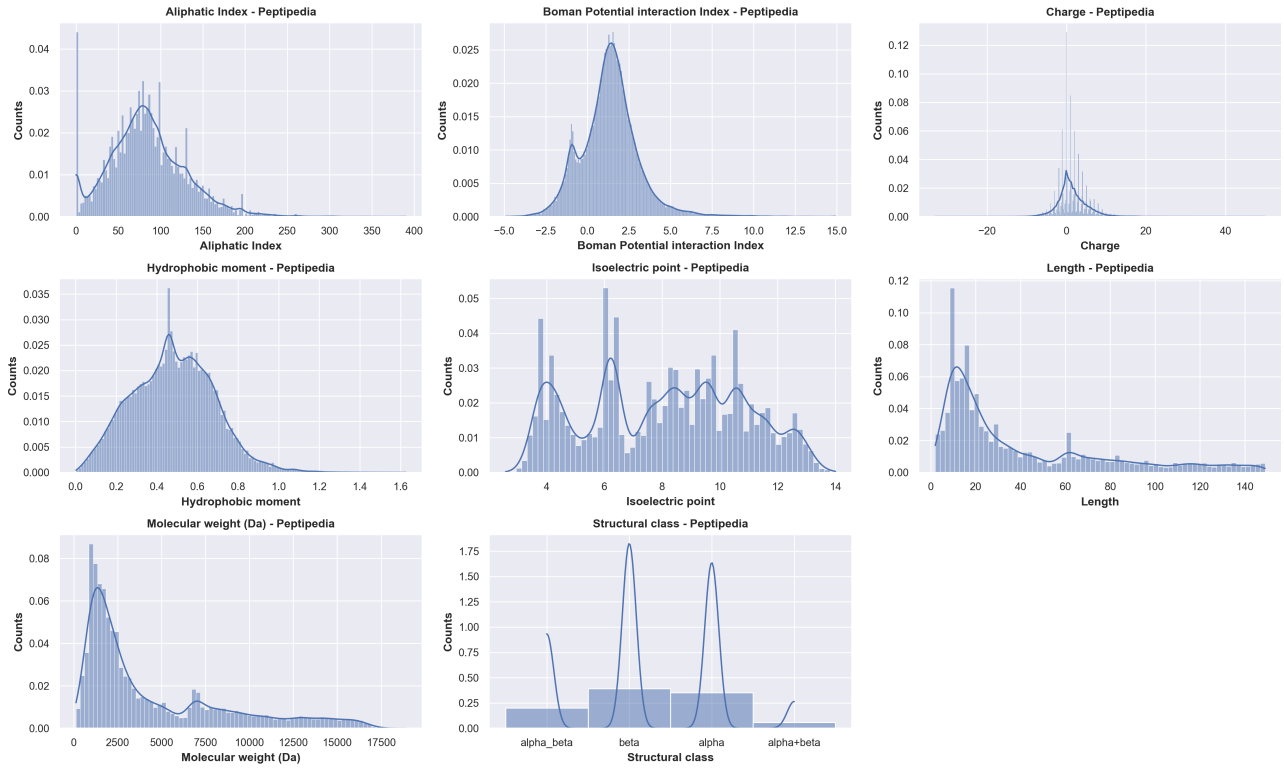

Fig. 1. Histograms describing the physicochemical properties of the Antiviral dataset. Curves represent the kernel density estimators of the different underlying distributions.

B. Search for Negative Peptides

Table S1 compiles the bioactivity tags excluded from the negative set when building the "NegSearch" datasets. The meaning behind these tags can be further expanded in the original publication (Quiroz et al., 2021).

Table 1. Overlapping classes excluded from the negative set for each of the benchmark datasets.

Dataset	Overlapping bioactivities
Antibacterial	`Antibacterial/antibiotic`
ACE inhibitor	`Blood_pressure,Blood_processes,Vasodilator,Vascular`
Anticancer	`Anicancer,Cytotoxic,Antitumour`
Antifungal	`Antifungal`
Antimalarial	`Antimalarial/antiplasmodial`
Antimicrobial	`Antimicrobial`
Antioxidant	`Antioxidant`
Antiparasitic	`Antiparasitic`
Antiviral	`Antiviral`
Blood-brain barrier	`Neuropeptide,Blood-brain_barrier_crossing`
DPPIV inhibitor	`Diabetic`
Anti-MRSA	`Antibacterial/antibiotic`
Neuropeptide	`Neuropeptide`
Quorum sensing	`Quorum_sensing`
Toxicity	`Cytotoxic,Neurotoxin,Toxic,Toxins`
Tumour T-cell antigens	`Immunological_activity`

C. Default hyperparameter search space

Table S2 describes the hyperparameter space defined for all experiments using the "Optimised ML ensemble".

Table 2. Default hyperparameter search space for the ensemble used throughout the paper.

Model	Trials	Hyperparameter search space			
		Name	**Type**	**Range**	**Log-scale**
KNN	10	K	integer	1-30	No
		Weights	categorical	uniform or distance	No
RFC	10	Max depth	integer	2-20	No
		Number of estimators	integer	10-100	No
LightGBM	10	Max depth	integer	1-30	Yes
		Number of leaves	integer	5-50	Yes
		Learning rate	float	10e-3 - 0.3	Yes

D. UniDL4BioPep model architecture

UniDL4BioPep (Du et al., 2023) computes the peptide-level representations using ESM2-8M in the same way as described in Methods, by averaging across all residue-level representations. It then uses a 1D-convolutional neural network (1D-CNN) to make the predictions. The architecture for this 1D-CNN are fixed and are described in Table S3.

Table 3. UniDL4BioPep architecture.

Layer	Type of layer	Input	Size	Output
1	Conv1D	320	32	32×320
2	MaxPool	32×320	$\sim$	32×160
3	Flatten	32×160	$\sim$	5,120
4	Dense	5,120	64	64
5	Output	64	2	2

E. Dataset diversity

The effect of the homology-based diversity of the training and testing subsets is represented in Table S4. Figures S2-S17 describe the physicochemical composition of all the datasets, showing the differences between positive and negative peptides.

Table 4. Number of connected components clusters per dataset. Original: refers to the dataset with original set of negatives; New: refers to the datasets with the new negatives.

Dataset	Original Training	NegSearch+HP Training	Original Test	NegSearch+HP Test
Antibacterial	1,339	3,288	6,642	2,229
Inhibitor of ACE enzyme	1,570	421	1,749	763
Anticancer	372	344	440	179
Antifungal	297	348	940	341
Antimalarial	173	55	1,389	278
Antimicrobial	1,087	2,276	9,880	6,827
Antioxidant	694	174	1,119	240
Antiparasitic	234	120	1,362	77
Antiviral	1,690	1,174	3,242	1,003
Brain-blood barrier	173	47	157	36
Inhibitor of DPPIV enzyme	1,014	265	868	261
Anti-MRSA	100	59	752	195
Neuropeptide	2,331	968	2,601	816
Quorum sensing	301	87	236	38
Toxicity	754	717	811	334
Tumor T-cell antigen	885	236	850	195

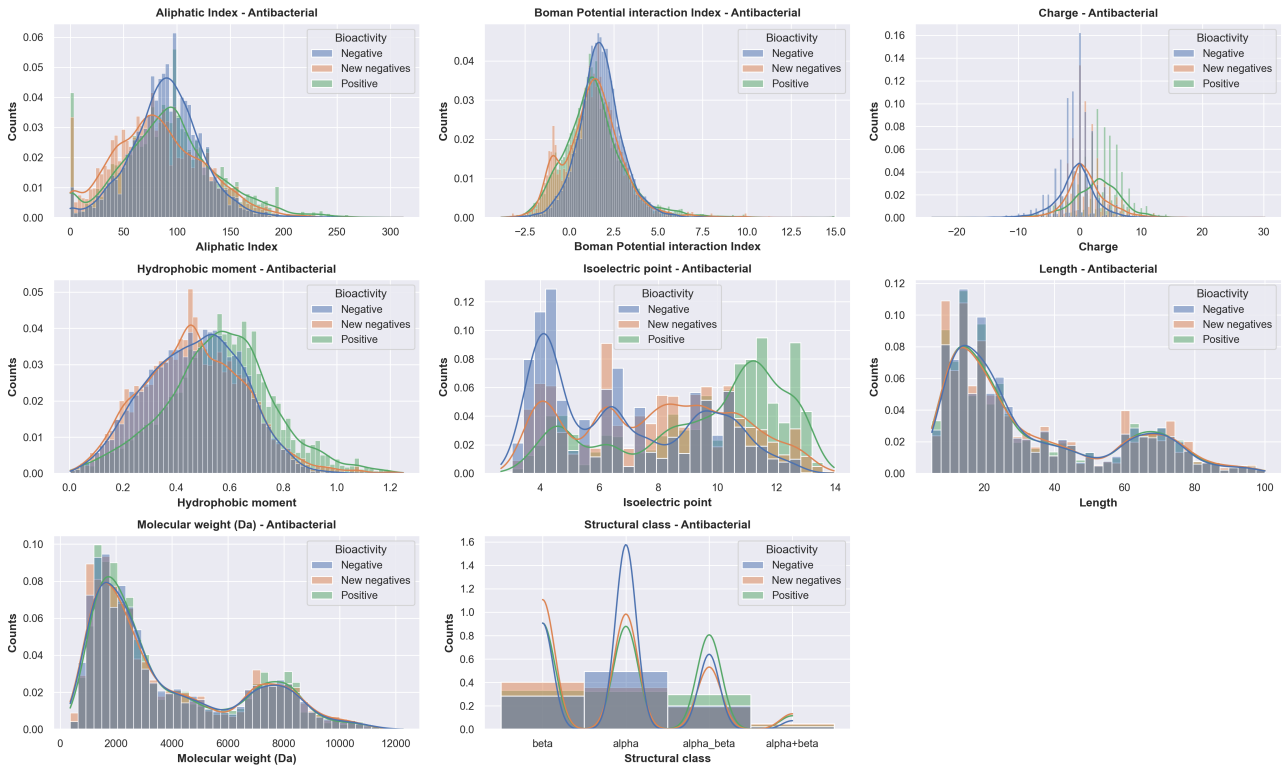

Fig. 2. Histograms describing the physicochemical properties of the Antibacterial dataset. Curves represent the kernel density estimators of the different underlying distributions.

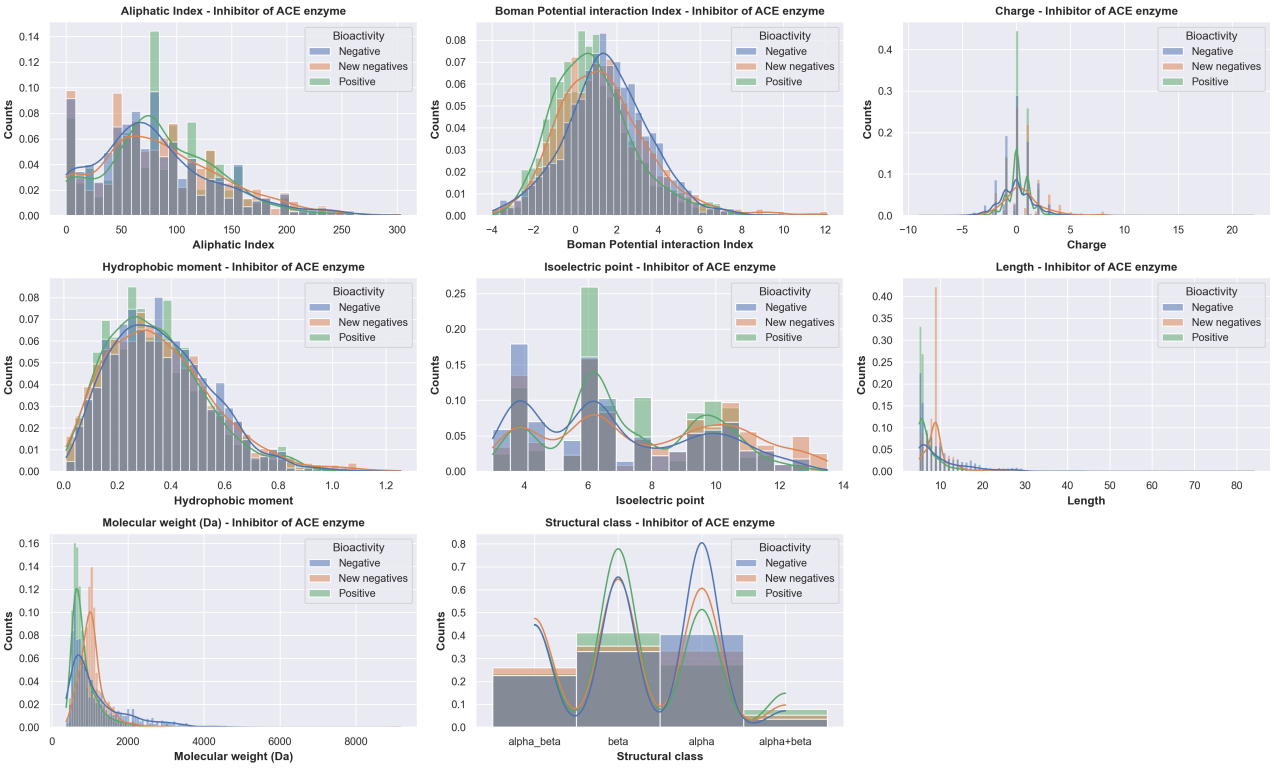

Fig. 3. Histograms describing the physicochemical properties of the ACE inhibitor dataset. Curves represent the kernel density estimators of the different underlying distributions.

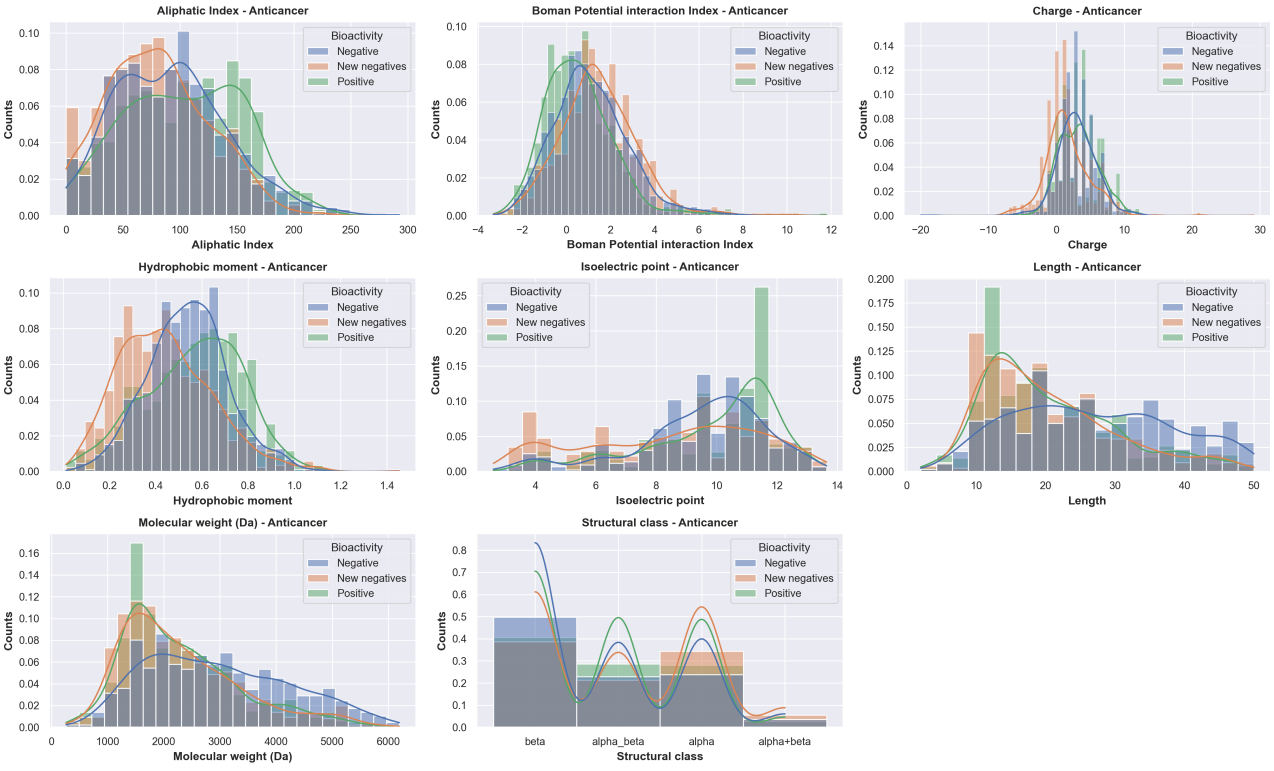

Fig. 4. Histograms describing the physicochemical properties of the Anticancer dataset. Curves represent the kernel density estimators of the different underlying distributions.

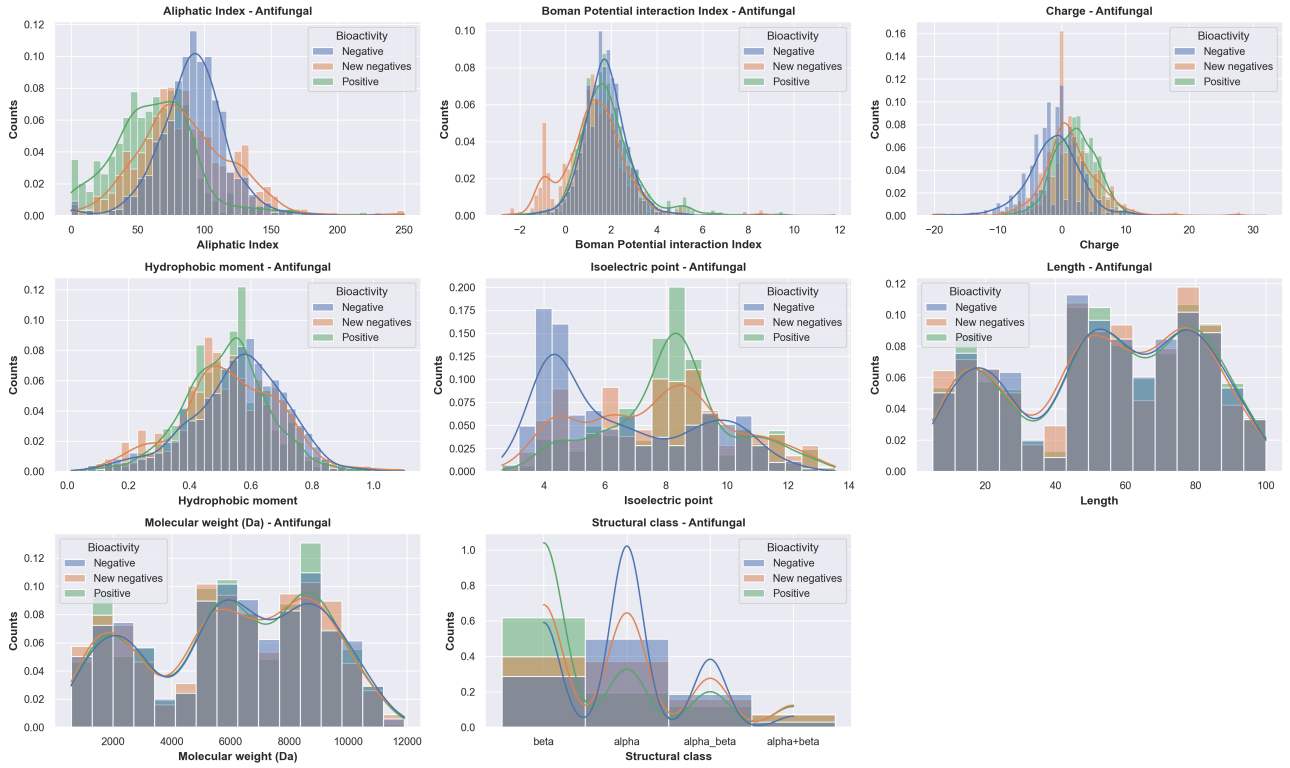

Fig. 5. Histograms describing the physicochemical properties of the Antifungal dataset. Curves represent the kernel density estimators of the different underlying distributions.

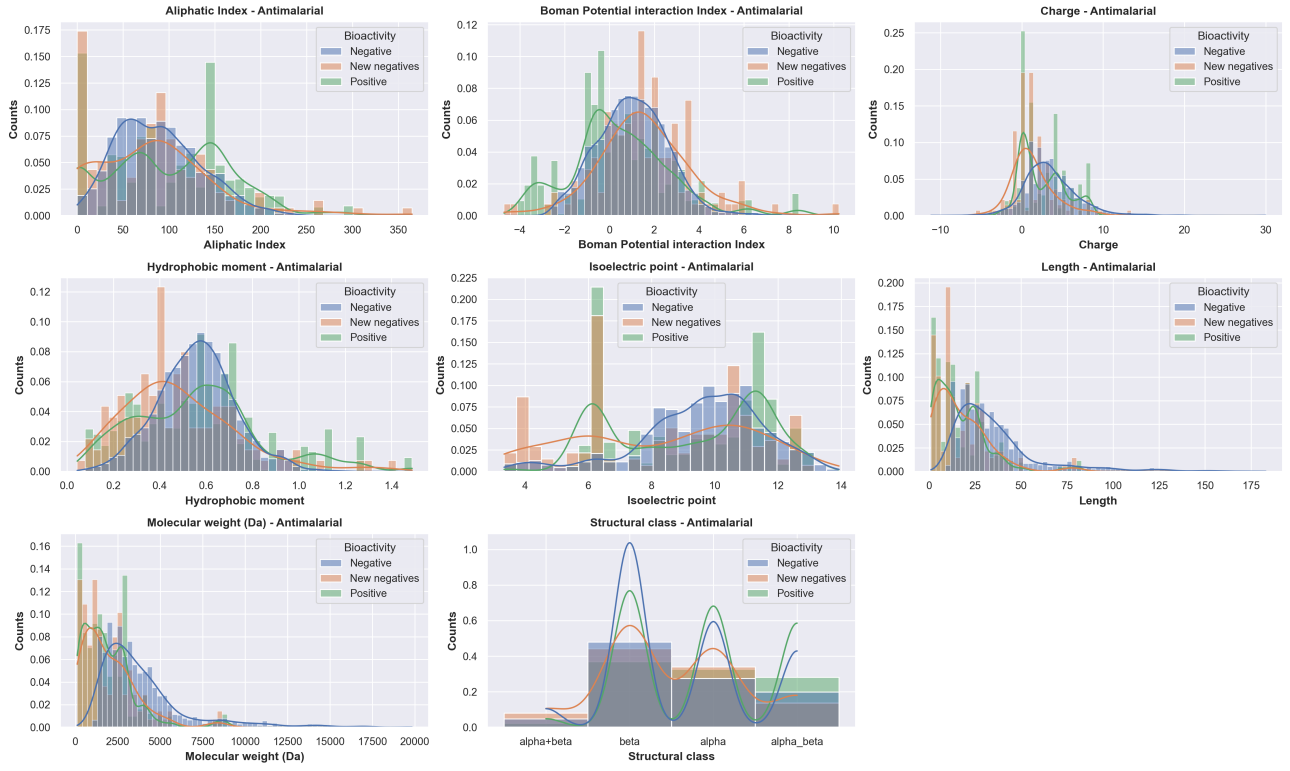

Fig. 6. Histograms describing the physicochemical properties of the Antimalarial dataset. Curves represent the kernel density estimators of the different underlying distributions.

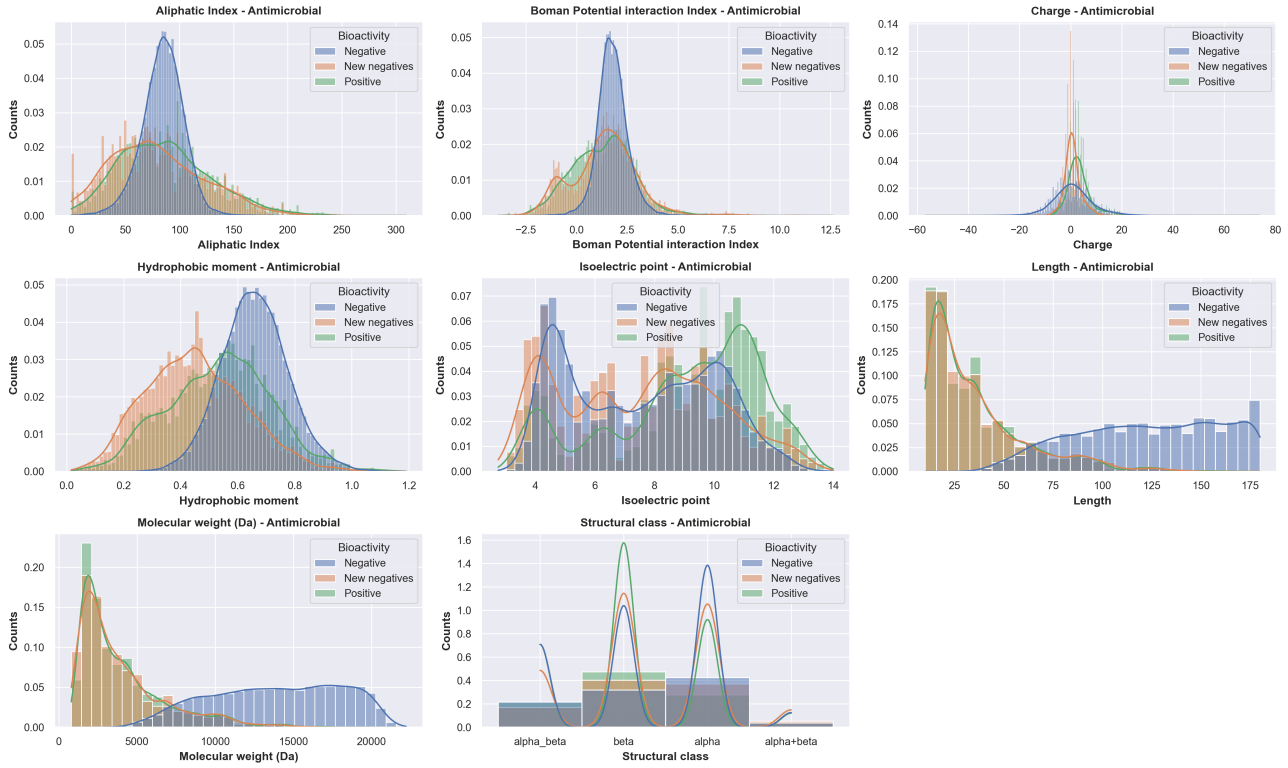

Fig. 7. Histograms describing the physicochemical properties of the Antimicrobial dataset. Curves represent the kernel density estimators of the different underlying distributions.

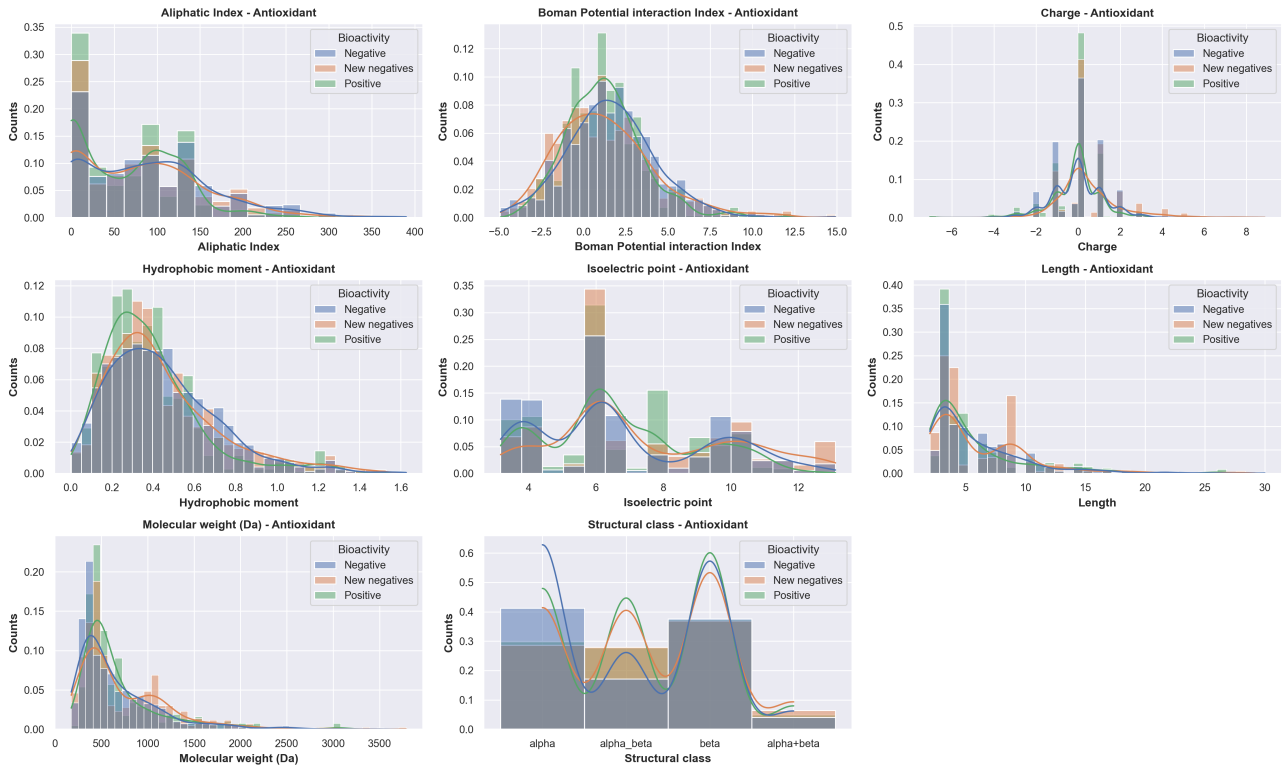

Fig. 8. Histograms describing the physicochemical properties of the Antioxidant dataset. Curves represent the kernel density estimators of the different underlying distributions.

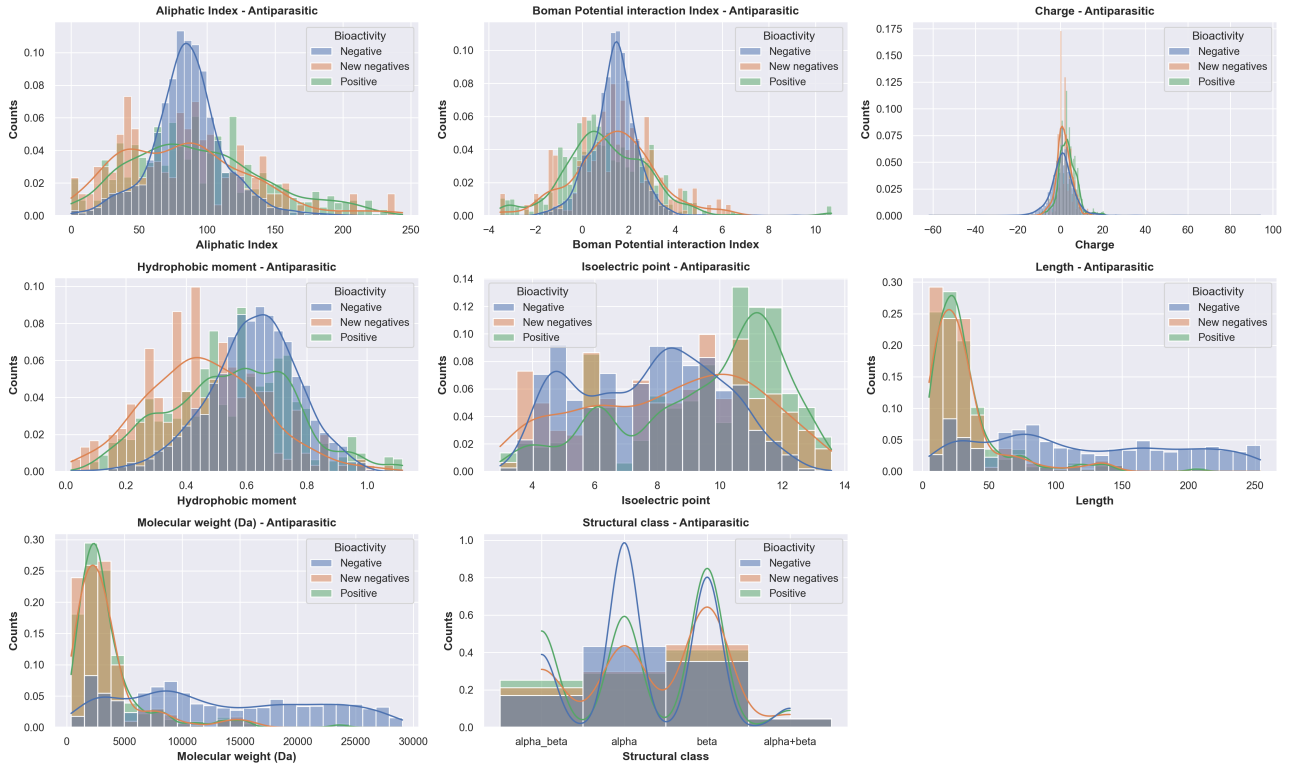

Fig. 9. Histograms describing the physicochemical properties of the Antiparasitic dataset. Curves represent the kernel density estimators of the different underlying distributions.

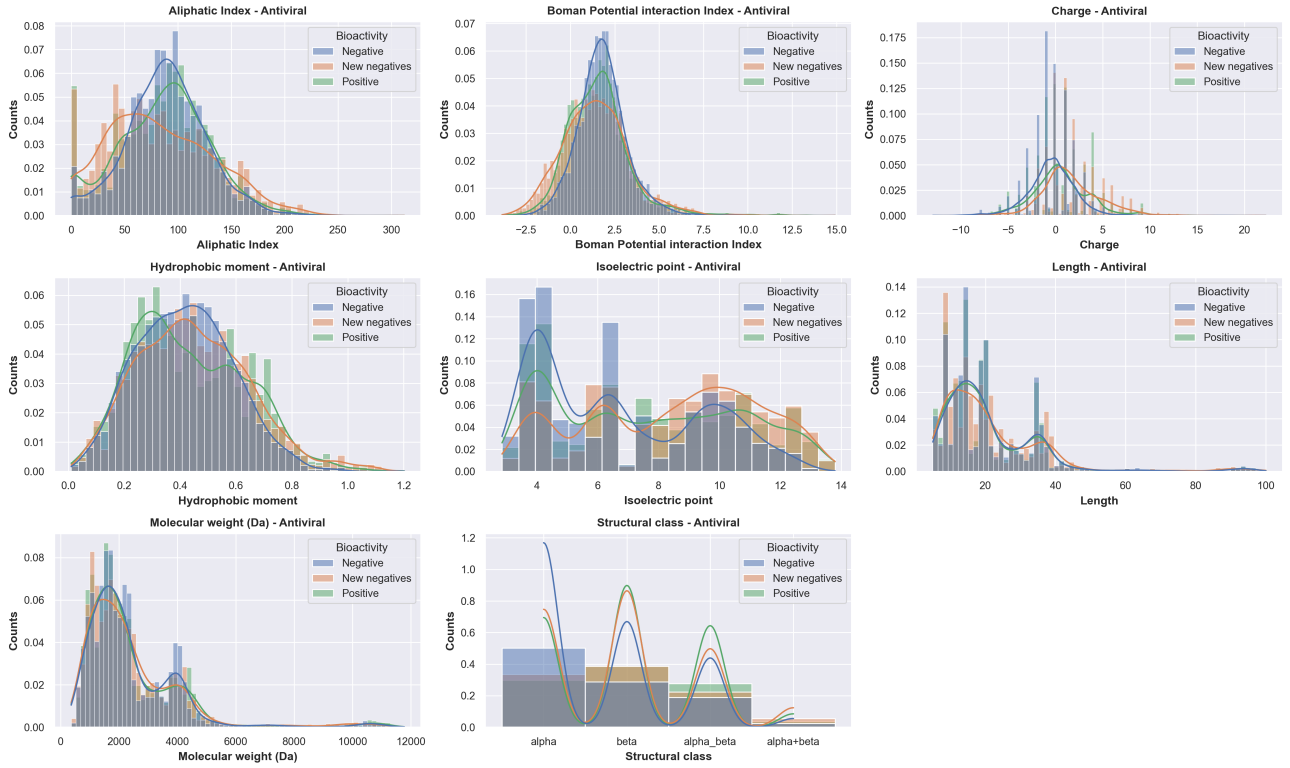

Fig. 10. Histograms describing the physicochemical properties of the Antiviral dataset. Curves represent the kernel density estimators of the different underlying distributions.

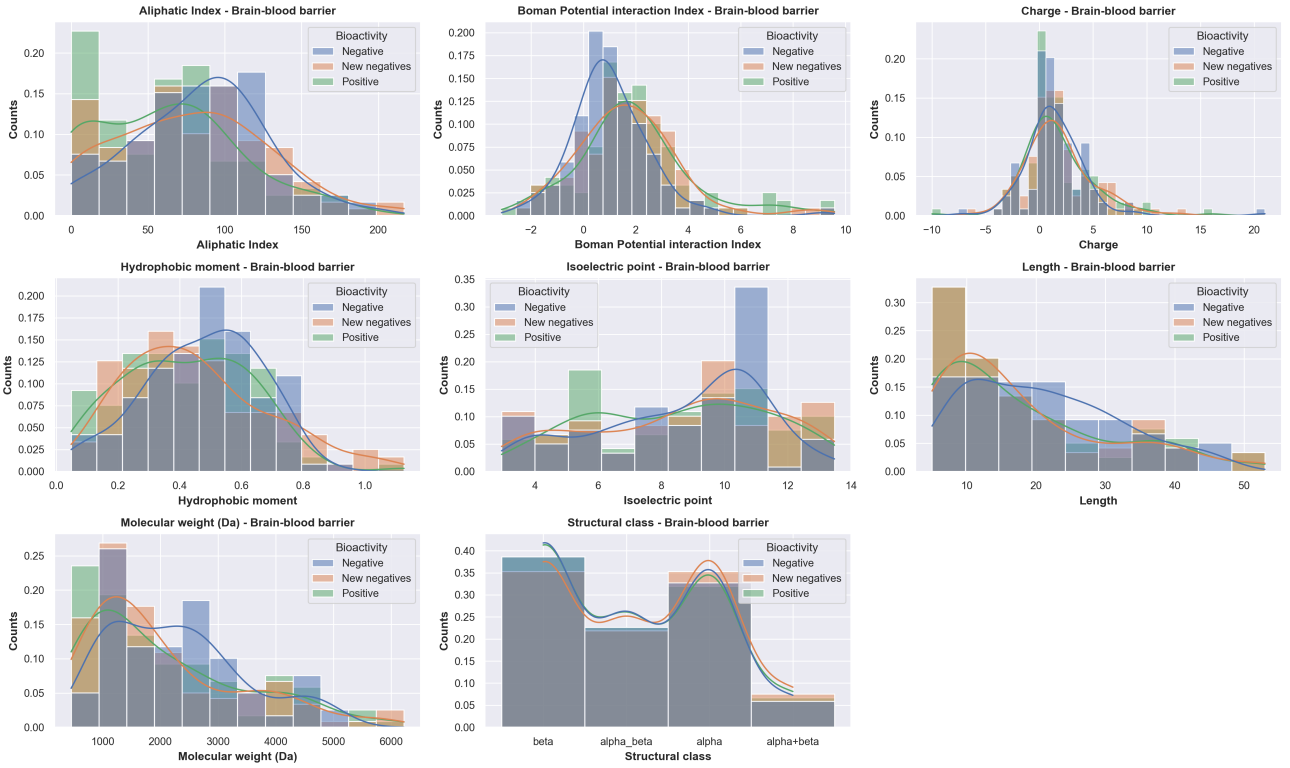

Fig. 11. Histograms describing the physicochemical properties of the Brain-blood barrier crossing dataset. Curves represent the kernel density estimators of the different underlying distributions.

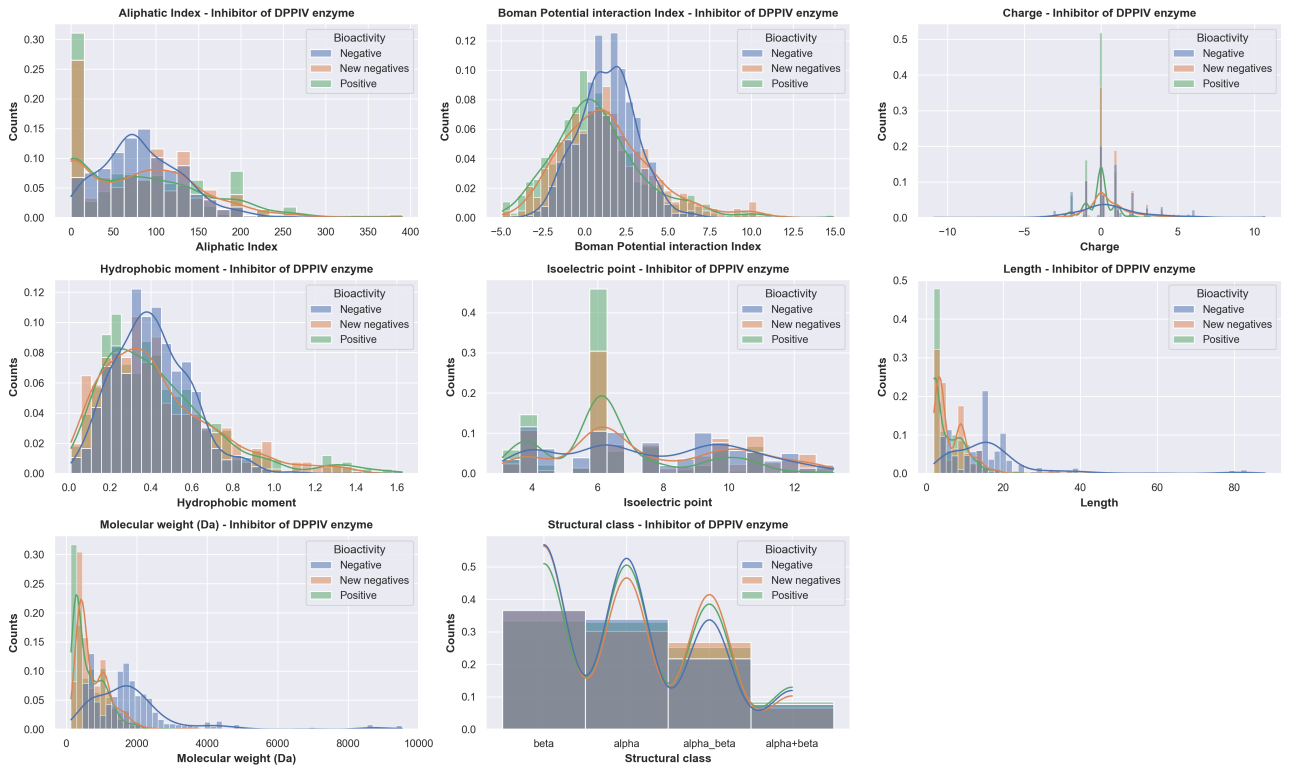

Fig. 12. Histograms describing the physicochemical properties of the DPPIV inhibitor dataset. Curves represent the kernel density estimators of the different underlying distributions.

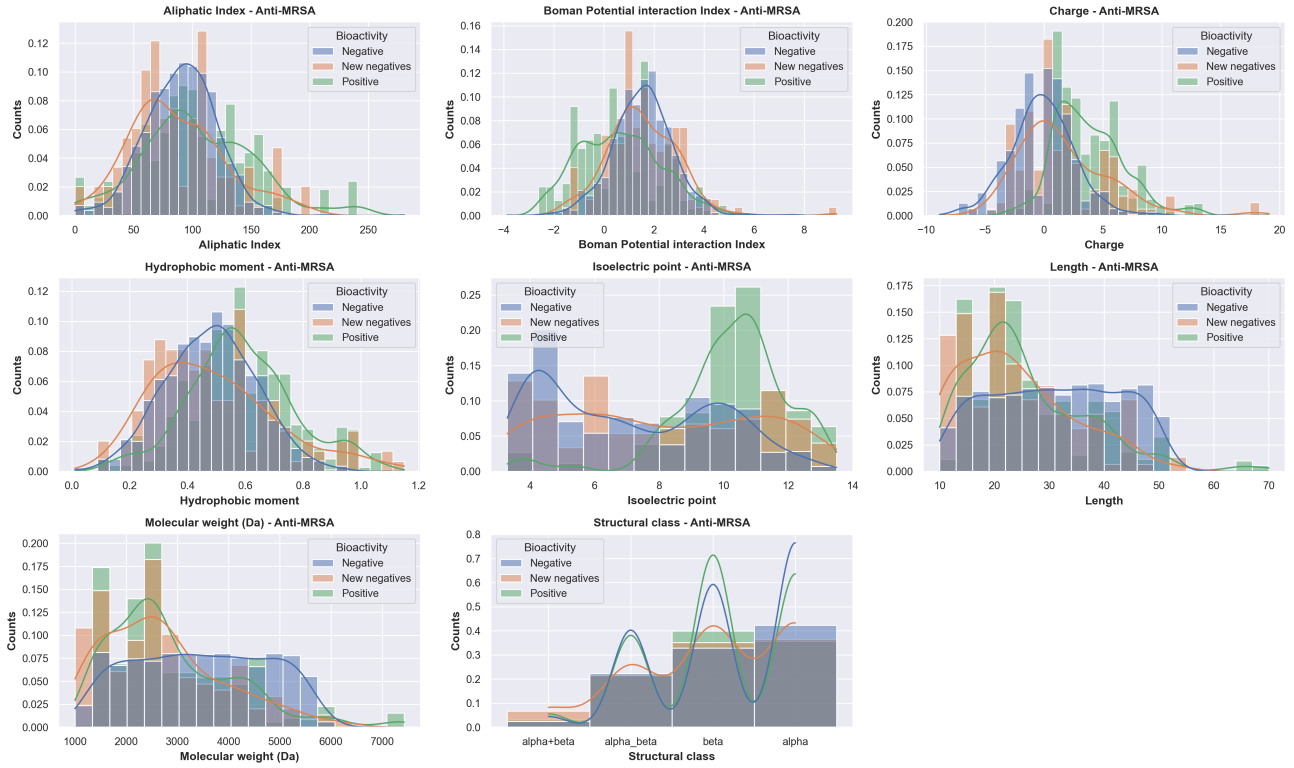

Fig. 13. Histograms describing the physicochemical properties of the Anti-MRSA dataset. Curves represent the kernel density estimators of the different underlying distributions.

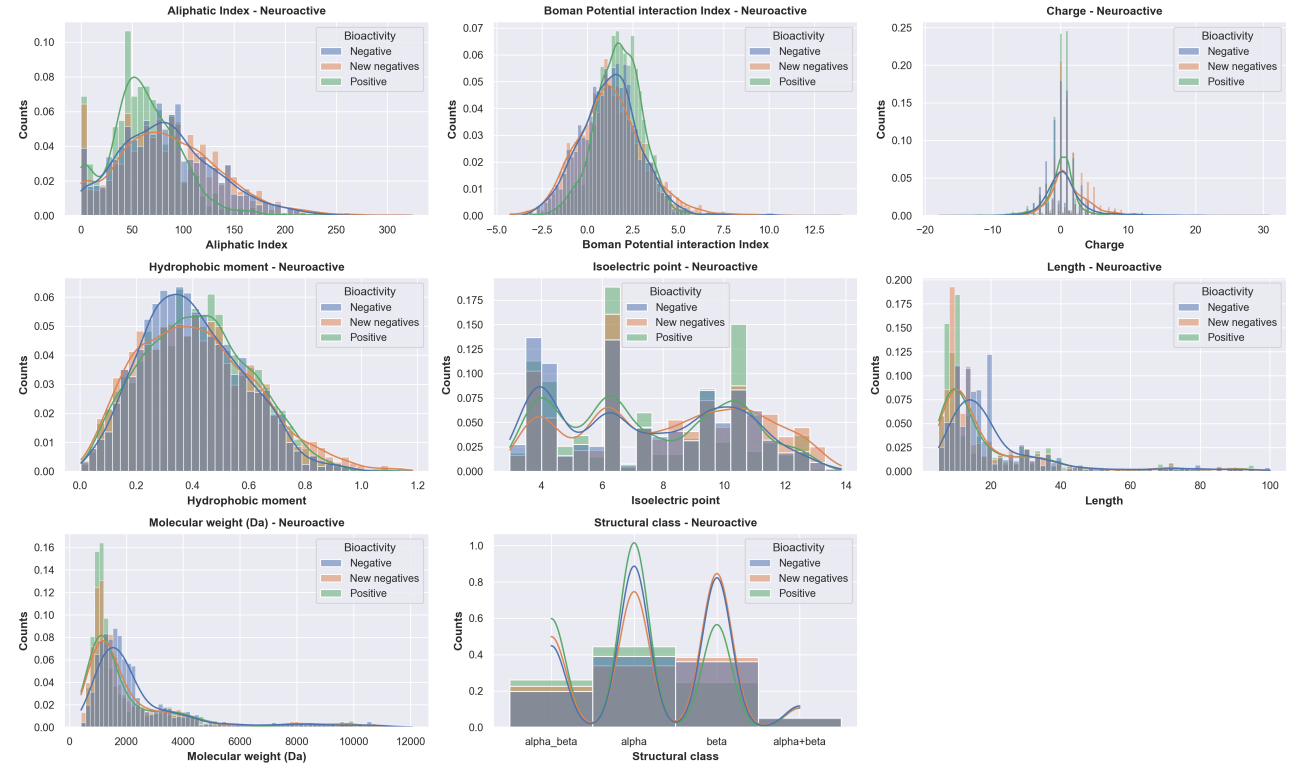

Fig. 14. Histograms describing the physicochemical properties of the Neuropeptide dataset. Curves represent the kernel density estimators of the different underlying distributions.

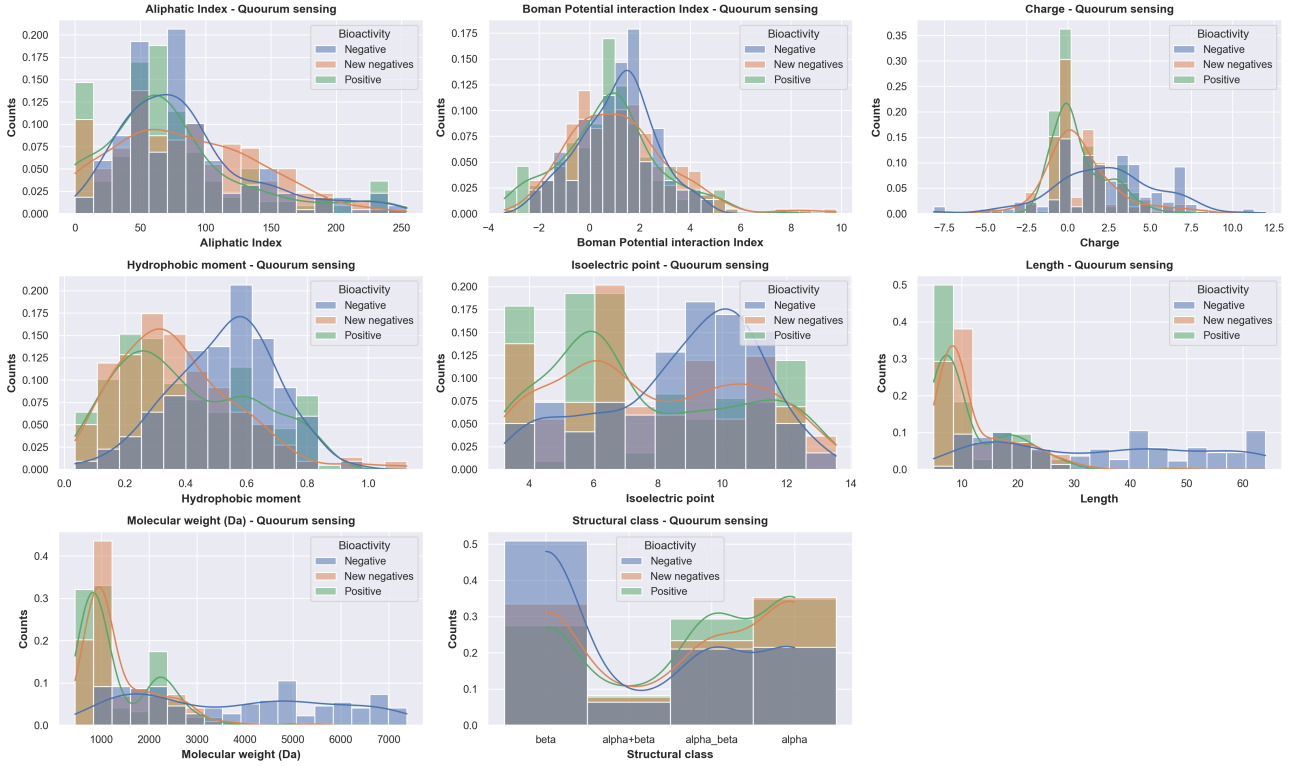

Fig. 15. Histograms describing the physicochemical properties of the Quorum sensing dataset. Curves represent the kernel density estimators of the different underlying distributions.

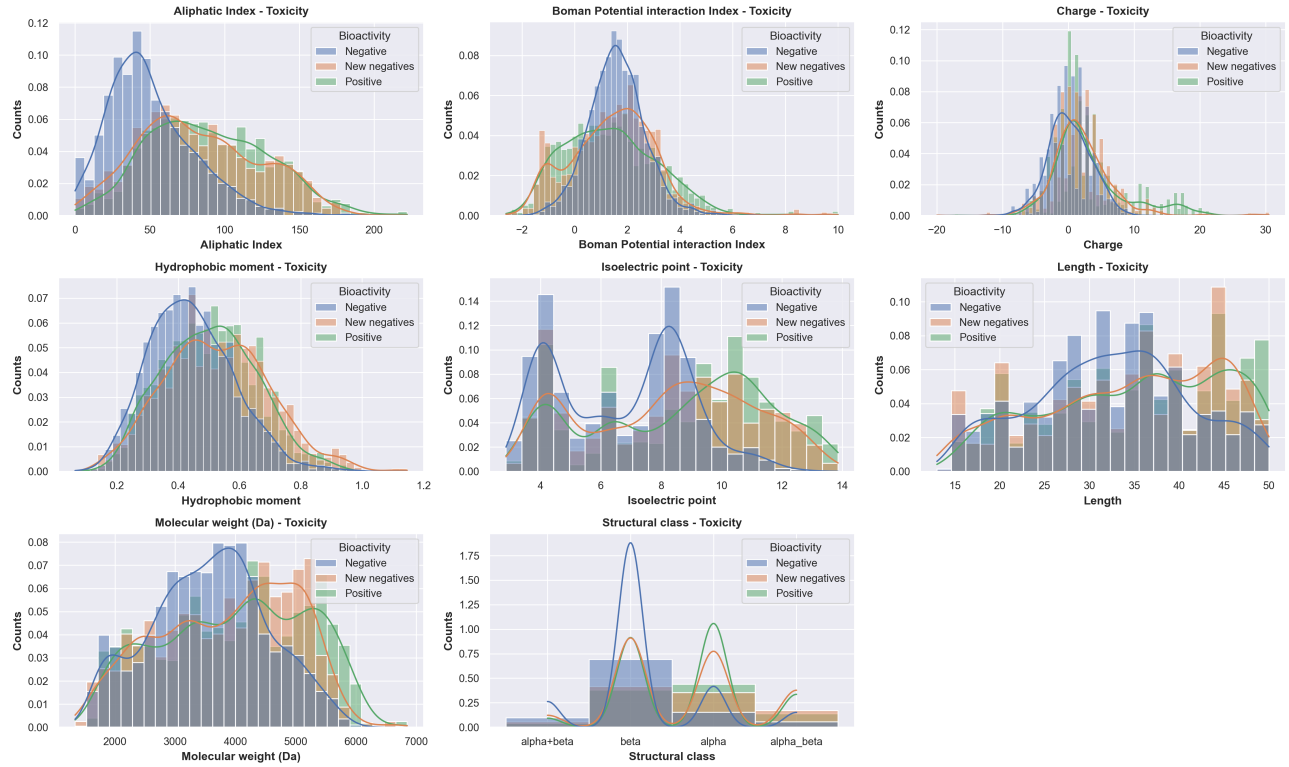

Fig. 16. Histograms describing the physicochemical properties of the Toxicity dataset. Curves represent the kernel density estimators of the different underlying distributions.

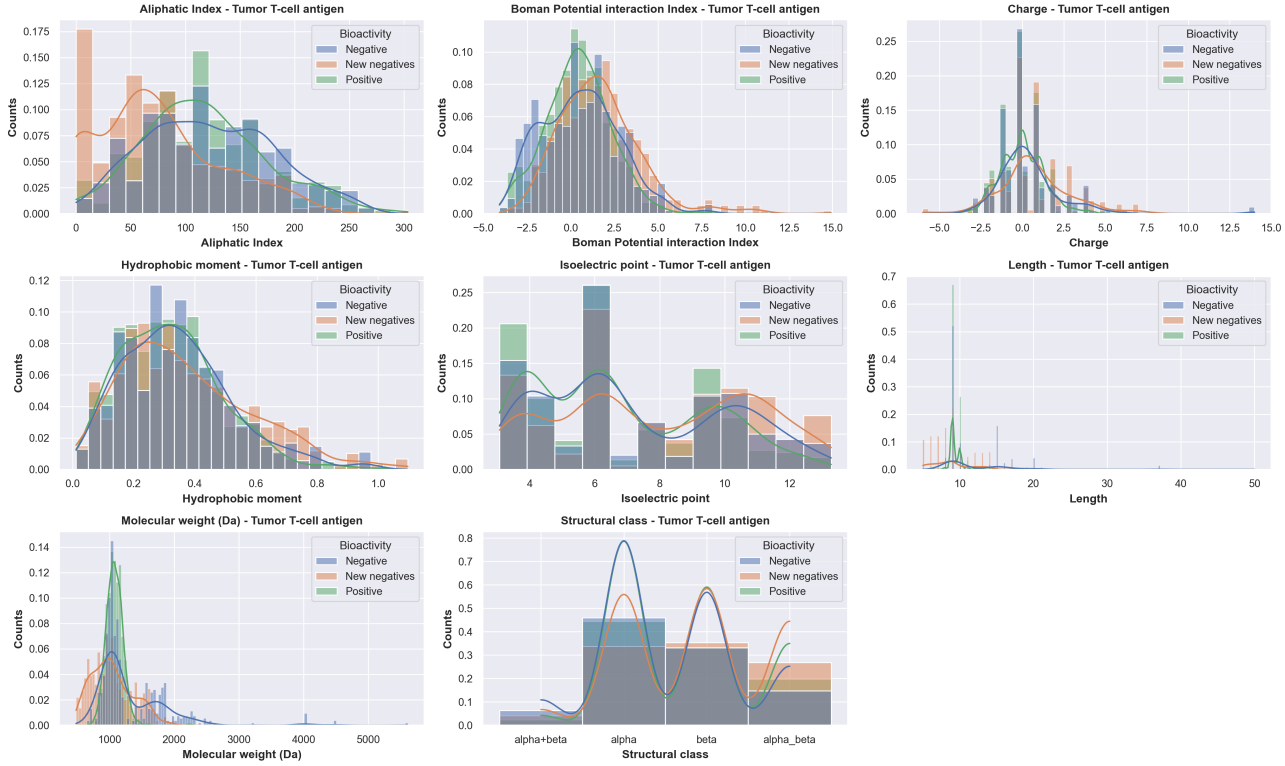

Fig. 17. Histograms describing the physicochemical properties of the Tumor T-cell antigen dataset. Curves represent the kernel density estimators of the different underlying distributions.

F. Metrics for model performance

This sections contains alternative metrics for all experimental results shown throughout the text.

Evaluation of AutoPeptideML's dataset construction modules

This subsection focuses on expanding on Figure 3. It includes Tables S5.

Table 5. Alternative metrics for the evaluation of AutoPeptideML's dataset construction modules. Errors represent the standard error of the mean across three different runs. Original: Original benchmark; NegSearch: Dataset with new negative peptides; HP: Homology-based dataset partitioning module; ACC: Accuracy; MCC: Mathew's correlation coeficient; AUROC: Area Under the ROC curve; F1: F1 score; BBBC: Brain-blood barrier crossing; TTCA: Tumor T-cell antigens.

Dataset	Metric	Original	NegSearch	NegSearch+HP
Antibacterial	ACC	0.9289 ± 0.0009	0.70 ± 0.02	0.717 ± 0.007
	MCC	0.858 ± 0.001	0.42 ± 0.02	0.45 ± 0.01
	AUROC	0.9770 ± 0.0006	0.767 ± 0.008	0.790 ± 0.005
	F1	0.926 ± 0.001	0.65 ± 0.03	0.67 ± 0.01
ACE inhibitor	ACC	0.855 ± 0.007	0.80 ± 0.01	0.783 ± 0.007
	MCC	0.71 ± 0.01	0.61 ± 0.03	0.57 ± 0.01
	AUROC	0.922 ± 0.008	0.87 ± 0.01	0.85 ± 0.01
	F1	0.856 ± 0.007	0.80 ± 0.01	0.786 ± 0.006
Anticancer	ACC	0.83 ± 0.05	0.63 ± 0.01	0.65 ± 0.01
	MCC	0.7 ± 0.1	0.30 ± 0.03	0.33 ± 0.01
	AUROC	0.88 ± 0.04	0.687 ± 0.006	0.707 ± 0.009
	F1	0.83 ± 0.05	0.55 ± 0.01	0.57 ± 0.03
Antifungal	ACC	0.947 ± 0.002	0.61 ± 0.01	0.594 ± 0.009
	MCC	0.895 ± 0.005	0.0.26 ± 0.02	0.20 ± 0.02
	AUROC	0.991 ± 0.002	0.63 ± 0.02	0.62 ± 0.02
	F1	0.944 ± 0.002	0.51 ± 0.02	0.49 ± 0.02
Antimalarial	ACC	0.984 ± 0.002	0.66 ± 0.02	0.69 ± 0.03
	MCC	0.89 ± 0.03	0.33 ± 0.04	0.39 ± 0.06
	AUROC	0.98 ± 0.01	0.74 ± 0.03	0.752 ± 0.003
	F1	0.90 ± 0.03	0.66 ± 0.04	0.0692 ± 0.009
Antimicrobial	ACC	0.953 ± 0.001	0.656 ± 0.008	0.645 ± 0.004
	MCC	0/887 ± 0.003	0.32 ± 0.01	0.292 ± 0.009
	AUROC	0.9860 ± 0.0008	0.736 ± 0.008	0.723 ± 0.007
	F1	0.919 ± 0.002	0.60 ± 0.02	0.628 ± 0.005
Antioxidant	ACC	0.83 ± 0.02	0.66 ± 0.02	0.63 ± 0.03
	MCC	0.67 ± 0.02	0.34 ± 0.05	0.26 ± 0.06
	AUROC	0.897 ± 0.002	0.71 ± 0.03	0.71 ± 0.03
	F1	0.814 ± 0.008	0.67 ± 0.04	0.64 ± 0.03
Antiparasitic	ACC	0.76 ± 0.02	0.700 ± 0.03	0.71 ± 0.04
	MCC	0.56 ± 0.02	0.40 ± 0.06	0.41 ± 0.07
	AUROC	0.930 ± 0.004	0.75 ± 0.03	0.76 ± 0.04
	F1	0.70 ± 0.03	0.69 ± 0.03	0.71 ± 0.03
Antiviral	ACC	0.828 ± 0.005	0.74 ± 0.02	0.760 ± 0.005
	MCC	0.659 ± 0.009	0.49 ± 0.05	0.520 ± 0.009
	AUROC	0.898 ± 0.005	0.82 ± 0.01	0.840 ± 0.005
	F1	0.821 ± 0.007	0.73 ± 0.03	0.764 ± 0.003
BBBC	ACC	0.80 ± 0.02	0.74 ± 0.02	0.54 ± 0.04
	MCC	0.60 ± 0.05	0.19 ± 0.06	0.08 ± 0.08
	AUROC	0.907 ± 0.008	0.65 ± 0.08	0.55 ± 0.06
	F1	0.79 ± 0.03	0.60 ± 0.03	0.0.56 ± 0.02
DPPIV inhibitor	ACC	0.83 ± 0.02	0.60 ± 0.03	0.74 ± 0.01
	MCC	0.67 ± 0.04	0.56 ± 0.04	0.47 ± 0.03
	AUROC	0.927 ± 0.002	0.84 ± 0.02	0.816 ± 0.008
	F1	0.83 ± 0.02	0.78 ± 0.02	0.73 ± 0.02
Anti-MRSA	ACC	0.998 ± 0.001	0.78 ± 0.02	0.60 ± 0.04
	MCC	0.993 ± 0.007	0.49 ± 0.03	0.41 ± 0.09
	AUROC	1.0000 ± 0.0000	0.82 ± 0.01	0.789 ± 0.009
	F1	0.994 ± 0.006	0.69 ± 0.02	0.65 ± 0.03
Neuropeptide	ACC	0.850 ± 0.002	0.853 ± 0.004	0.817 ± 0.007
	MCC	0.705 ± 0.003	0.708 ± 0.008	0.64 ± 0.01
	AUROC	0.937 ± 0.001	0.919 ± 0.002	0.908 ± 0.004
	F1	0.858 ± 0.002	0.852 ± 0.004	0.830 ± 0.05
Quorum sensing	ACC	0.908 ± 0.008	0.839 ± 0.007	0.82 ± 0.04
	MCC	0.82 ± 0.02	0.68 ± 0.01	0.65 ± 0.08
	AUROC	0.967 ± 0.005	0.934 ± 0.008	0.93 ± 0.02
	F1	0.908 ± 0.008	0.842 ± 0.005	0.83 ± 0.03
Toxicity	ACC	0.914 ± 0.004	0.684 ± 0.006	0.628 ± 0.003
	MCC	0.828 ± 0.009	0.39 ± 0.01	0.26 ± 0.006
	AUROC	0.9677 ± 0.0007	0.756 ± 0.001	0.736 ± 0.005
	F1	0.919 ± 0.004	0.727 ± 0.006	0.670 ± 0.004
TTCA	ACC	0.689 ± 0.009	0.900 ± 0.008	0.869 ± 0.009
	MCC	0.33 ± 0.02	0.80 ± 0.02	0.74 ± 0.02
	AUROC	0.70 ± 0.01	0.95 ± 0.02	0.93 ± 0.01
	F1	0.753 ± 0.008	0.899 ± 0.007	0.877 ± 0.007

Evaluation of different protein language models

This subsection focuses on expanding on Figure 3. It includes Tables S6-S7.

Table 6. Alternative metrics for the evaluation for different protein language models. Errors represent the standard error of the mean across three different runs. ACC: Accuracy; MCC: Mathew's correlation coeficient; AUROC: Area Under the ROC curve; F1: F1 score; BBBC: Brain-blood barrier crossing; TTCA: Tumor T-cell antigens.

Dataset	Metric	ESM2 8M	ESM2 35M	ESM2 150M	ESM2 650M	ESM1b 650M	ProtBERT	Prot-T5-XL	Prost-T5
Antibacterial	ACC	0.717 ± 0.007	0.720 ± 0.002	0.726 ± 0.002	0.719 ± 0.002	0.699 ± 0.005	0.711 ± 0.001	0.70 ± 0.01	0.67 ± 0.02
	MCC	0.45 ± 0.01	0.457 ± 0.005	0.465 ± 0.003	0.459 ± 0.005	0.42 ± 0.01	0.43 ± 0.02	0.41 ± 0.02	0.35 ± 0.04
	AUROC	0.790 ± 0.005	0.794 ± 0.003	0.799 ± 0.005	0.796 ± 0.003	0.793 ± 0.005	0.780 ± 0.007	0.787 ± 0.006	0.754 ± 0.009
	F1	0.67 ± 0.01	0.677 ± 0.002	0.689 ± 0.002	0.670 ± 0.005	0.65 ± 0.01	0.679 ± 0.01	0.639 ± 0.02	0.60 ± 0.04
ACE inhibitor	ACC	0.783 ± 0.007	0.79 ± 0.01	0.77 ± 0.01	0.776 ± 0.002	0.79 ± 0.02	0.72 ± 0.01	0.7561 ± 0.0008	0.7593 ± 0.007
	MCC	0.57 ± 0.1	0.58 ± 0.02	0.53 ± 0.03	0.552 ± 0.003	0.589 ± 0.03	0.44 ± 0.02	0.513 ± 0.001	0.52 ± 0.01
	AUROC	0.85 ± 0.01	0.85 ± 0.01	0.84 ± 0.01	0.854 ± 0.007	0.86 ± 0.02	0.78 ± 0.02	0.833 ± 0.005	0.83 ± 0.01
	F1	0.786 ± 0.006	0.79 ± 0.01	0.78 ± 0.01	0.776 ± 0.005	0.79 ± 0.01	0.72 ± 0.01	0.760 ± 0.004	0.764 ± 0.006
Anticancer	ACC	0.65 ± 0.01	0.65 ± 0.01	0.64 ± 0.01	0.0612 ± 0.009	0.616 ± 0.004	0.63 ± 0.01	0.62 ± 0.01	0.620 ± 0.007
	MCC	0.33 ± 0.02	0.33 ± 0.02	0.30 ± 0.03	0.26 ± 0.02	0.27 ± 0.01	0.28 ± 0.02	0.26 ± 0.03	0.25 ± 0.02
	AUROC	0.707 ± 0.009	0.73 ± 0.02	0.73 ± 0.01	0.73 ± 0.02	0.72 ± 0.02	0.71 ± 0.02	0.71 ± 0.01	0.680 ± 0.009
	F1	0.57 ± 0.03	0.57 ± 0.04	0.54 ± 0.05	0.47 ± 0.02	0.472 ± 0.007	0.56 ± 0.01	0.51 ± 0.03	0.546 ± 0.004
Antifungal	ACC	0.594 ± 0.009	0.62 ± 0.01	0.64 ± 0.01	0.651 ± 0.009	0.60 ± 0.02	0.62 ± 0.02	0.749 ± 0.007	0.611 ± 0.01
	MCC	0.20 ± 0.02	0.25 ± 0.04	0.28 ± 0.03	0.33 ± 0.02	0.22 ± 0.05	0.23 ± 0.04	0.31 ± 0.01	0.23 ± 0.03
	AUROC	0.62 ± 0.02	0.67 ± 0.01	0.71 ± 0.02	0.72 ± 0.01	0.69 ± 0.02	0.68 ± 0.03	0.72 ± 0.01	0.66 ± 0.2
	F1	0.49 ± 0.02	0.564 ± 0.009	0.587 ± 0.004	0.56 ± 0.02	0.517 ± 0.008	0.60 ± 0.02	0.59 ± 0.02	0.547 ± 0.009
Antimalarial	ACC	0.63 ± 0.03	0.63 ± 0.03	0.66 ± 0.04	0.72 ± 0.03	0.69 ± 0.02	0.648 ± 0.03	0.67 ± 0.02	0.70 ± 0.05
	MCC	0.39 ± 0.06	0.27 ± 0.06	0.32 ± 0.07	0.43 ± 0.07	0.38 ± 0.04	0.30 ± 0.05	0.33 ± 0.03	0.4 ± 0.1
	AUROC	0.752 ± 0.003	0.72 ± 0.02	0.72 ± 0.03	0.80 ± 0.01	0.74 ± 0.02	0.72 ± 0.04	0.70 ± 0.02	0.74 ± 0.03
	F1	0.692 ± 0.009	0.64 ± 0.04	0.66 ± 0.04	0.72 ± 0.03	0.68 ± 0.02	0.61 ± 0.04	0.62 ± 0.02	0.68 ± 0.06
Antimicrobial	ACC	0.645 ± 0.005	0.6494 ± 0.0009	0.649 ± 0.005	0.64 ± 0.1	0.638 ± 0.009	0.619 ± 0.004	0.659 ± 0.009	0.640 ± 0.006
	MCC	0.292 ± 0.009	0.300 ± 0.002	0.30 ± 0.01	0.29 ± 0.02	0.28 ± 0.02	0.239 ± 0.008	0.32 ± 0.02	0.28 ± 0.01
	AUROC	0.723 ± 0.007	0.722 ± 0.004	0.723 ± 0.004	0.731 ± 0.007	0.723 ± 0.007	0.682 ± 0.007	0.74 ± 0.01	0.737 ± 0.008
	F1	0.628 ± 0.005	0.632 ± 0.003	0.627 ± 0.005	0.62 ± 0.01	0.61 ± 0.01	0.617 ± 0.005	0.644 ± 0.009	0.621 ± 0.005
Antioxidant	ACC	0.63 ± 0.03	0.632 ± 0.008	0.62 ± 0.03	0.63 ± 0.03	0.66 ± 0.03	0.59 ± 0.04	0.068 ± 0.03	0.65 ± 0.03
	MCC	0.27 ± 0.06	0.29 ± 0.02	0.25 ± 0.02	0.27 ± 0.06	0.31 ± 0.06	0.18 ± 0.07	0.37 ± 0.06	0.31 ± 0.05
	AUROC	0.71 ± 0.03	0.71 ± 0.02	0.69 ± 0.04	0.70 ± 0.02	0.72 ± 0.03	0.66 ± 0.04	0.74 ± 0.05	0.70 ± 0.03
	F1	0.64 ± 0.03	0.66 ± 0.01	0.63 ± 0.03	0.66 ± 0.03	0.65 ± 0.04	0.59 ± 0.04	0.69 ± 0.04	0.66 ± 0.03
Antiparasitic	ACC	0.71 ± 0.03	0.70 ± 0.02	0.69 ± 0.02	0.700 ± 0.01	0.70 ± 0.03	0.71 ± 0.04	0.714 ± 0.003	0.69 ± 0.02
	MCC	0.41 ± 0.07	0.41 ± 0.04	0.37 ± 0.04	0.40 ± 0.02	0.41 ± 0.06	0.43 ± 0.07	0.428 ± 0.006	0.37 ± 0.03
	AUROC	0.76 ± 0.04	0.77 ± 0.03	0.77 ± 0.03	0.79 ± 0.02	0.78 ± 0.04	0.80 ± 0.04	0.79 ± 0.01	0.77 ± 0.02
	F1	0.71 ± 0.03	0.70 ± 0.02	0.716 ± 0.004	0.715 ± 0.004	0.71 ± 0.02	0.73 ± 0.03	0.715 ± 0.005	0.69 ± 0.02
Antiviral	ACC	0.760 ± 0.005	0.739 ± 0.003	0.737 ± 0.004	0.749 ± 0.008	0.745 ± 0.008	0.69 ± 0.02	0.747 ± 0.009	0.710 ± 0.007
	MCC	0.520 ± 0.009	0.470 ± 0.005	0.474 ± 0.008	0.50 ± 0.02	0.49 ± 0.02	0.38 ± 0.03	0.50 ± 0.02	0.42 ± 0.01
	AUROC	0.840 ± 0.005	0.811 ± 0.003	0.819 ± 0.002	0.825 ± 0.005	0.821 ± 0.006	0.77 ± 0.01	0.831 ± 0.05	0.80 ± 0.01
	F1	0.764 ± 0.003	0.745 ± 0.002	0.742 ± 0.004	0.760 ± 0.006	0.751 ± 0.008	0.70 ± 0.01	0.749 ± 0.007	0.712 ± 0.007
BBBC	ACC	0.54 ± 0.04	0.61 ± 0.05	0.56 ± 0.01	0.60 ± 0.01	0.62 ± 0.02	0.60 ± 0.03	0.72 ± 0.03	0.60 ± 0.09
	MCC	0.08 ± 0.08	0.22 ± 0.1	0.12 ± 0.03	0.21 ± 0.02	0.25 ± 0.04	0.21 ± 0.05	0.43 ± 0.05	0.19 ± 0.2
	AUROC	0.55 ± 0.06	0.58 ± 0.07	0.627 ± 0.006	0.64 ± 0.03	0.66 ± 0.03	0.64 ± 0.04	0.74 ± 0.02	0.64 ± 0.08
	F1	0.56 ± 0.02	0.61 ± 0.04	0.55 ± 0.5	0.60 ± 0.03	0.64 ± 0.02	0.57 ± 0.03	0.72 ± 0.03	0.59 ± 0.09
DPPIV inhibitor	ACC	0.75 ± 0.01	0.76 ± 0.02	0.731 ± 0.003	0.766 ± 0.008	0.76 ± 0.01	0.75 ± 0.01	0.77 ± 0.01	0.757 ± 0.007
	MCC	0.47 ± 0.03	0.52 ± 0.04	0.462 ± 0.007	0.53 ± 0.02	0.52 ± 0.02	0.50 ± 0.02	0.54 ± 0.03	0.52 ± 0.01
	AUROC	0.815 ± 0.008	0.84 ± 0.01	0.81 ± 0.1	0.834 ± 0.006	0.83 ± 0.01	0.82 ± 0.02	0.860 ± 0.006	0.836 ± 0.008
	F1	0.73 ± 0.02	0.76 ± 0.02	0.733 ± 0.007	0.77 ± 0.01	0.75 ± 0.02	0.75 ± 0.02	0.77 ± 0.02	0.75 ± 0.01
Anti-MRSA	ACC	0.69 ± 0.04	0.73 ± 0.01	0.64 ± 0.03	0.69 ± 0.04	0.68 ± 0.02	0.71 ± 0.04	0.66 ± 0.03	0.71 ± 0.03
	MCC	0.41 ± 0.09	0.48 ± 0.01	0.31 ± 0.08	0.42 ± 0.09	0.41 ± 0.06	0.43 ± 0.09	0.34 ± 0.07	0.44 ± 0.05
	AUROC	0.779 ± 0.009	0.81 ± 0.2	0.727 ± 0.005	0.81 ± 0.02	0.79 ± 0.03	0.80 ± 0.04	0.74 ± 0.03	0.76 ± 0.04
	F1	0.65 ± 0.03	0.70 ± 0.02	0.54 ± 0.04	0.60 ± 0.07	0.622 ± 0.002	0.685 ± 0.05	0.61 ± 0.05	0.67 ± 0.04
Neuropeptide	ACC	0.817 ± 0.007	0.82 ± 0.02	0.81 ± 0.01	0.833 ± 0.01	0.829 ± 0.007	0.77 ± 0.01	0.823 ± 0.003	0.78 ± 0.02
	MCC	0.64 ± 0.01	0.65 ± 0.03	0.63 ± 0.02	0.67 ± 0.02	0.66 ± 0.01	0.55 ± 0.02	0.654 ± 0.005	0.58 ± 0.03
	AUROC	0.908 ± 0.004	0.90 ± 0.01	0.90 ± 0.01	0.918 ± 0.008	0.905 ± 0.007	0.86 ± 0.01	0.912 ± 0.006	0.865 ± 0.01
	F1	0.830 ± 0.005	0.83 ± 0.01	0.82 ± 0.01	0.84 ± 0.01	0.839 ± 0.005	0.786 ± 0.009	0.836 ± 0.002	0.80 ± 0.01
Quorum sensing	ACC	0.82 ± 0.04	0.82 ± 0.03	0.82 ± 0.01	0.84 ± 0.01	0.85 ± 0.01	0.75 ± 0.04	0.831 ± 0.004	0.83 ± 0.03
	MCC	0.65 ± 0.08	0.64 ± 0.05	0.65 ± 0.02	0.67 ± 0.02	0.71 ± 0.03	0.51 ± 0.08	0.686 ± 0.003	0.66 ± 0.05
	AUROC	0.93 ± 0.02	0.92 ± 0.02	0.920 ± 0.006	0.938 ± 0.008	0.93 ± 0.02	0.85 ± 0.04	0.934 ± 0.004	0.90 ± 0.02
	F1	0.83 ± 0.03	0.82 ± 0.02	0.823 ± 0.003	0.836 ± 0.007	0.85 ± 0.02	0.76 ± 0.03	0.842 ± 0.001	0.84 ± 0.02
Toxicity	ACC	0.63 ± 0.01	0.627 ± 0.007	0.629 ± 0.003	0.63 ± 0.02	0.638 ± 0.007	0.62 ± 0.01	0.68 ± 0.01	0.640 ± 0.002
	MCC	0.264 ± 0.006	0.26 ± 0.01	0.264 ± 0.007	0.28 ± 0.03	0.28 ± 0.02	0.25 ± 0.03	0.36 ± 0.02	0.288 ± 0.004
	AUROC	0.736 ± 0.005	0.726 ± 0.009	0.721 ± 0.003	0.74 ± 0.01	0.74 ± 0.01	0.715 ± 0.009	0.793 ± 0.005	0.733 ± 0.004
	F1	0.670 ± 0.004	0.672 ± 0.005	0.666 ± 0.002	0.68 ± 0.02	0.674 ± 0.006	0.651 ± 0.01	0.706 ± 0.009	0.679 ± 0.004
TTCA	ACC	0.869 ± 0.009	0.86 ± 0.01	0.85 ± 0.01	0.87 ± 0.01	0.86 ± 0.01	0.82 ± 0.02	0.85 ± 0.02	0.8386 ± 0.02
	MCC	0.74 ± 0.02	0.72 ± 0.02	0.71 ± 0.02	0.74 ± 0.03	0.73 ± 0.02	0.64 ± 0.04	0.70 ± 0.03	0.74 ± 0.05
	AUROC	0.93 ± 0.01	0.92 ± 0.02	0.92 ± 0.01	0.926 ± 0.008	0.929 ± 0.008	0.88 ± 0.02	0.92 ± 0.02	0.93 ± 0.0.009
	F1	0.877 ± 0.007	0.87 ± 0.01	0.86 ± 0.01	0.87 ± 0.01	0.87 ± 0.01	0.83 ± 0.01	0.86 ± 0.01	0.8884 ± 0.02

Comparison of training strategies on original datasets

This subsection contains alternative metrics for the results in Figure 4.A. Figures S18-S35 represent the ROC curves corresponding to one of the three experiments. Table S8 contains alternative metrics for the comparison of training strategies on the original datasets.

Alternative metrics for the comparison of training strategies in the new datasets.

This subsection reflects the results in Figure 4.B. It includes Table S9.

Table 7. Metrics for one-hot encoding.

Dataset	MCC
Antibacterial	0.293 ± 0.007
ACE inhibitor	0.40 ± 0.06
Anticancer	0.270 ± 0.02
Antifungal	0.06 ± 0.03
Antimalarial	0.32 ± 0.08
Antimicrobial	0.076 ± 0.007
Antioxidant	0.239 ± 0.02
Antiparasitic	0.16 ± 0.04
Antiviral	0.385 ± 0.04
BBBC	0.39 ± 0.01
DPPIV inhibitor	0.40 ± 0.05
Anti-MRSA	0.32 ± 0.03
Neuropeptide	0.53 ± 0.04
Quorum sensing	0.41 ± 0.03
Toxicity	0.08 ± 0.03
TTCA	0.56 ± 0.01

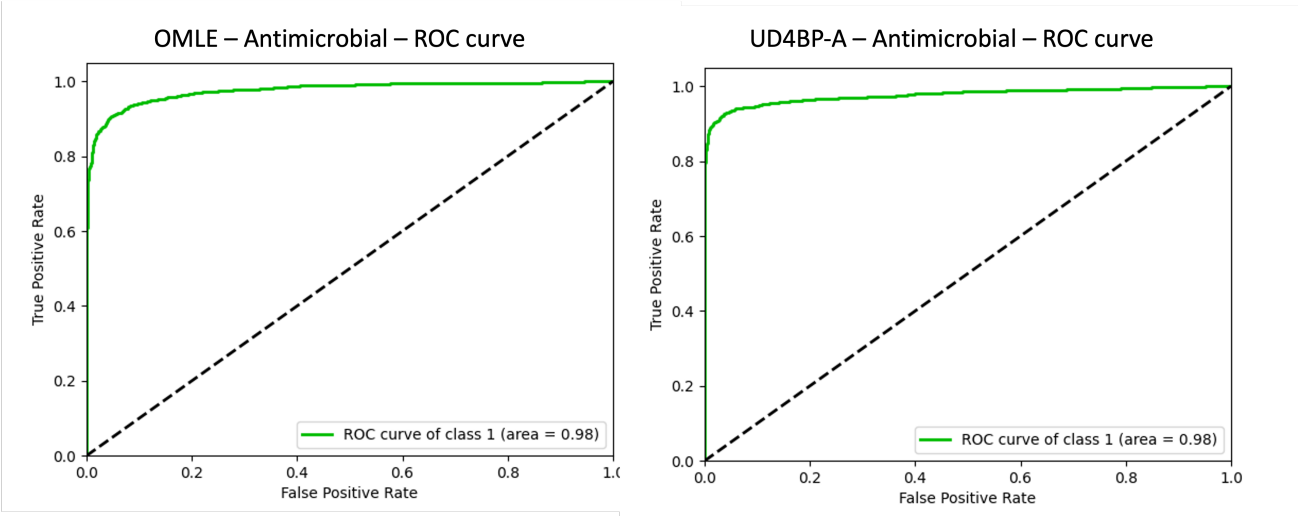

Fig. 18. ROC curves of the original Antibacterial dataset. OMLE (left) and UniDL4BioPep (right).

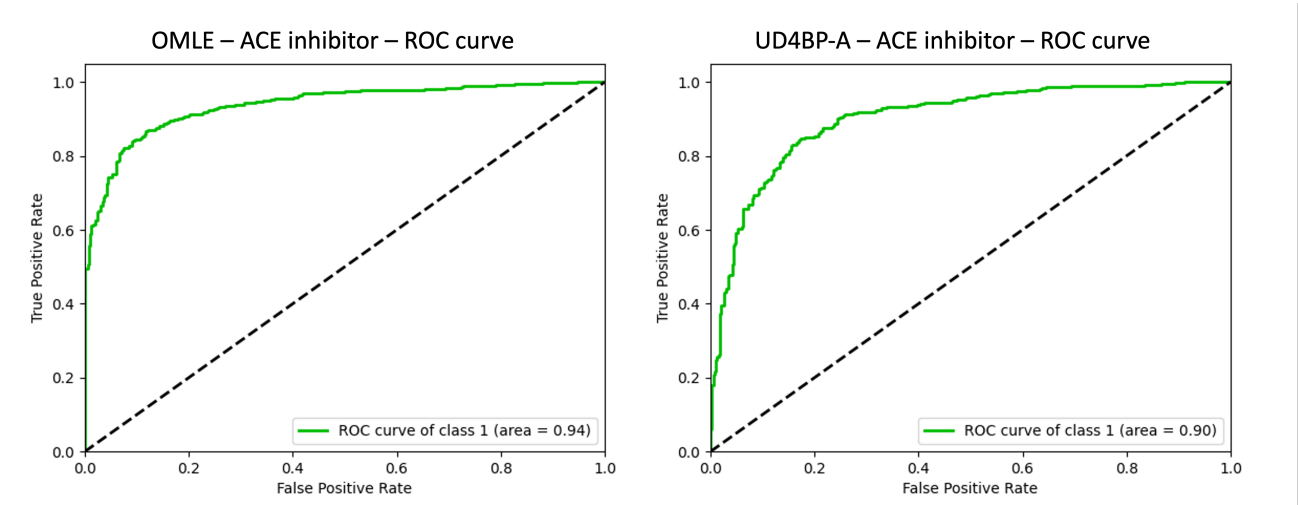

Fig. 19. ROC curves of the original ACE inhibitor dataset. OMLE (left) and UniDL4BioPep (right).

Table 8. Alternative metrics for the comparison of training strategies on original datasets. Errors represent the standard error of the mean across three different runs. ACC: Accuracy; MCC: Mathew's correlation coeficient; AUROC: Area Under the ROC curve; F1: F1 score; BBBC: Brain-blood barrier crossing; TTCA: Tumor T-cell antigens; OMLE: Optimised Machine Learning Ensemble; UDL4BP-A: UnidDLBioPep-A. For the reference of the handcrafted models, see Table

Dataset	Metric	OMLE	UDL4BP-A	Handcrafted models
Antibacterial	ACC	0.92803 ± 0.0009	0.9427 ± 0.0008	0.935
	MCC	0.858 ± 0.001	0.887 ± 0.001	0.870
	AUROC	0.9770 ± 0.0005	0.9766 ± 0.0004	0.975
	F1	0.941 ± 0.001	0.677 ± 0.002	NA
ACE inhibitor	ACC	0.855 ± 0.007	0.9280 ± 0.0009	0.883
	MCC	0.71 ± 0.01	0.67 ± 0.01	0.767
	AUROC	0.922 ± 0.008	0.911 ± 0.005	0.951
	F1	0.856 ± 0.007	0.83 ± 0.01	NA
Anticancer 1	ACC	0.717 ± 0.009	0.749 ± 0.007	0.825
	MCC	0.43 ± 0.02	0.50 ± 0.01	0.646
	AUROC	0.800 ± 0.002	0.8106 ± 0.0006	0.812
	F1	0.714 ± 0.009	0.751 ± 0.006	NA
Anticancer 2	ACC	0.939 ± 0.002	0.841 ± 0.003	0.9201
	MCC	0.879 ± 0.005	0.883 ± 0.005	0.84
	AUROC	0.9685 ± 0.0005	0.969 ± 0.001	NA
	F1	0.937 ± 0.02	0.939 ± 0.003	NA
Antifungal	ACC	0.947 ± 0.002	0.947 ± 0.002	0.942
	MCC	0.895 ± 0.004	0.894 ± 0.005	0.884
	AUROC	0.991 ± 0.002	0.9875 ± 0.0001	0.988
	F1	0.944 ± 0.002	0.946 ± 0.002	NA
Antimalarial 1	ACC	0.980 ± 0.001	0.975 ± 0.004	0.978
	MCC	0.78 ± 0.03	0.82 ± 0.01	0.776
	AUROC	0.955 ± 0.003	0.935 ± 0.003	0.82
	F1	0.828 ± 0.009	0.79 ± 0.03	NA
Antimalarial 2	ACC	0.98770 ± 0.00001	0.973 ± 0.004	0.957
	MCC	0.95660 ± 0.00001	0.91 ± 0.01	0.834
	AUROC	0.997 ± 0.001	0.993 ± 0.001	0.903
	F1	0.96300 ± 0.00001	0.92 ± 0.01	NA
Antimicrobial	ACC	0.954 ± 0.001	0.9667 ± 0.007	NA
	MCC	0.887 ± 0.003	0.919 ± 0.002	NA
	AUROC	0.9860 ± 0.0008	0.9895 ± 0.0004	NA
	F1	0.919 ± 0.002	0.942 ± 0.001	NA
Antioxidant	ACC	0.84 ± 0.01	0.831 ± 0.004	NA
	MCC	0.67 ± 0.02	0.657 ± 0.08	0.48
	AUROC	0.897 ± 0.002	0.880 ± 0.07	0.79
1.	F1	0.814 ± 0.008	0.809 ± 0.005	NA
Antiparasitic	ACC	0.757 ± 0.01	0.754 ± 0.01	0.880
	MCC	0.56 ± 0.02	0.55 ± 0.02	0.776
	AUROC	0.930 ± 0.004	0.931 ± 0.006	0.922
	F1	0.70 ± 0.03	0.70 ± 0.02	0.891
Antiviral	ACC	0.828 ± 0.005	0.835 ± 0.001	0.828
	MCC	0.659 ± 0.009	0.673 ± 0.004	0.662
	AUROC	0.898 ± 0.005	0.9083 ± 0.0007	0.896
	F1	0.821 ± 0.007	0.829 ± 0.005	NA
BBBC	ACC	0.80 ± 0.02	0.78950 ± 0.00001	0.7895
	MCC	0.60 ± 0.05	0.58220 ± 0.00001	0.6102
	AUROC	0.907 ± 0.008	0.85 ± 0.02	0.7895
	F1	0.79 ± 0.03	0.77780 ± 0.00001	0.7500
DPPIV inhibitor	ACC	0.83 ± 0.02	0.812 ± 0.004	0.797
	MCC	0.67 ± 0.04	0.624 ± 0.009	0.594
	AUROC	0.927 ± 0.002	0.911 ± 0.003	0.847
	F1	0.83 ± 0.02	0.811 ± 0.004	NA
Anti-MRSA	ACC	0.998 ± 0.002	0.988 ± 0.004	0.960
	MCC	0.993 ± 0.007	0.96 ± 0.02	0.848
	AUROC	1.00000 ± 0.00001	0.9994 ± 0.0002	0.986
	F1	0.994 ± 0.006	0.96 ± 0.01	NA
Neuropeptide	ACC	0.850 ± 0.002	0.888 ± 0.002	0.936
	MCC	0.705 ± 0.003	0.776 ± 0.004	0.875
	AUROC	0.937 ± 0.001	0.953 ± 0.001	0.988
	F1	0.858 ± 0.002	0.889 ± 0.001	NA
Quorum sensing	ACC	0.908 ± 0.008	0.917 ± 0.008	0.943
	MCC	0.82 ± 0.01	0.83 ± 0.02	0.885
	AUROC	0.967 ± 0.005	0.981 ± 0.001	0.945
	F1	0.908 ± 0.008	0.915 ± 0.008	NA
Toxicity	ACC	0.914 ± 0.004	0.913 ± 0.004	0.912 ± 0.002
	MCC	0.828 ± 0.098	0.828 ± 0.005	0.903 ± 0.004
	AUROC	0.9677 ± 0.0007	0.9708 ± 0.0006	0.976 ± 0.001
	F1	0.919 ± 0.004	0.916 ± 0.002	NA
TTCA	ACC	0.689 ± 0.009	0.72 ± 0.01	0.71
	MCC	0.33 ± 0.02	0.42 ± 0.02	0.363
	AUROC	0.70 ± 0.01	0.782 ± 0.004	0.73
	F1	0.753 ± 0.008	0.77 ± 0.01	0.756

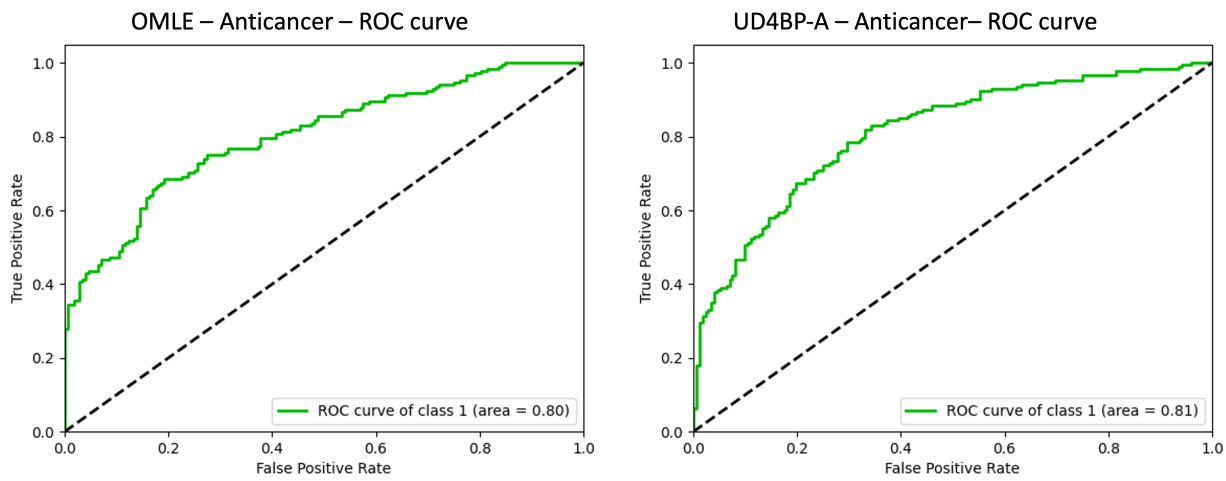

Fig. 20. ROC curves of the original Anticancer 1 dataset. OMLE (left) and UniDL4BioPep (right).

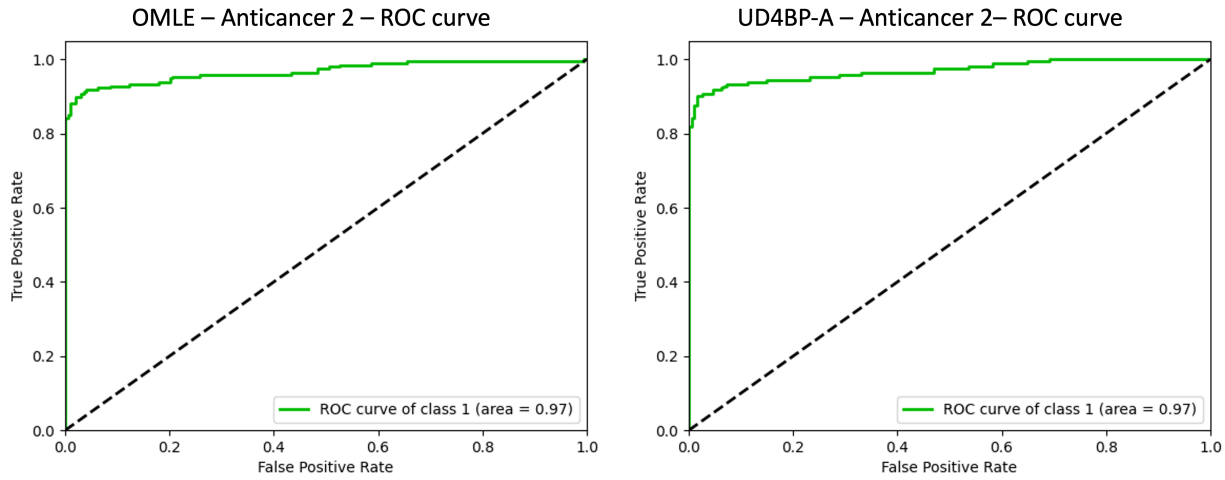

Fig. 21. ROC curves of the original Anticancer 2 dataset. OMLE (left) and UniDL4BioPep (right).

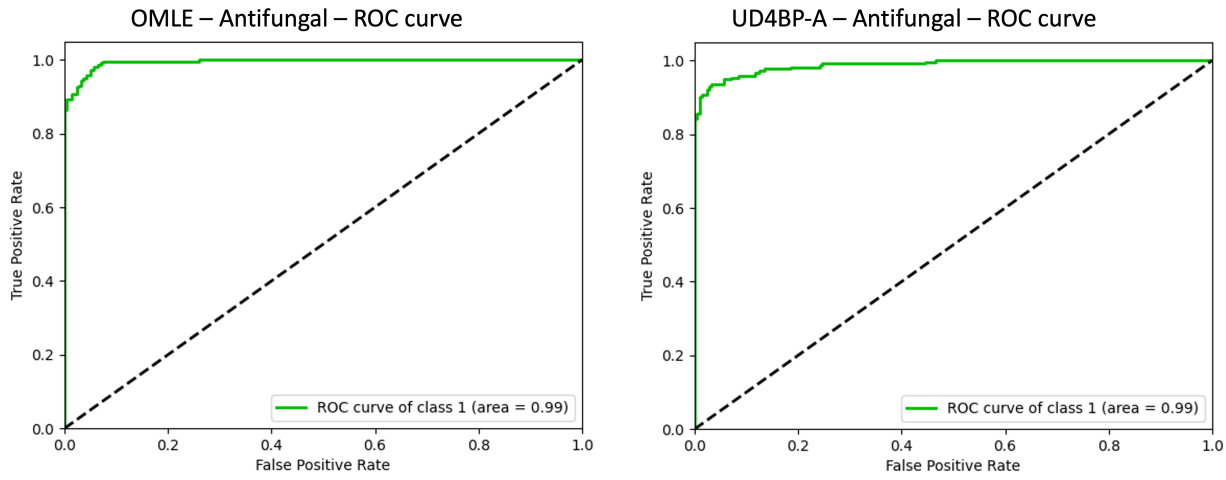

Fig. 22. ROC curves of the original Antifungal dataset. OMLE (left) and UniDL4BioPep (right).

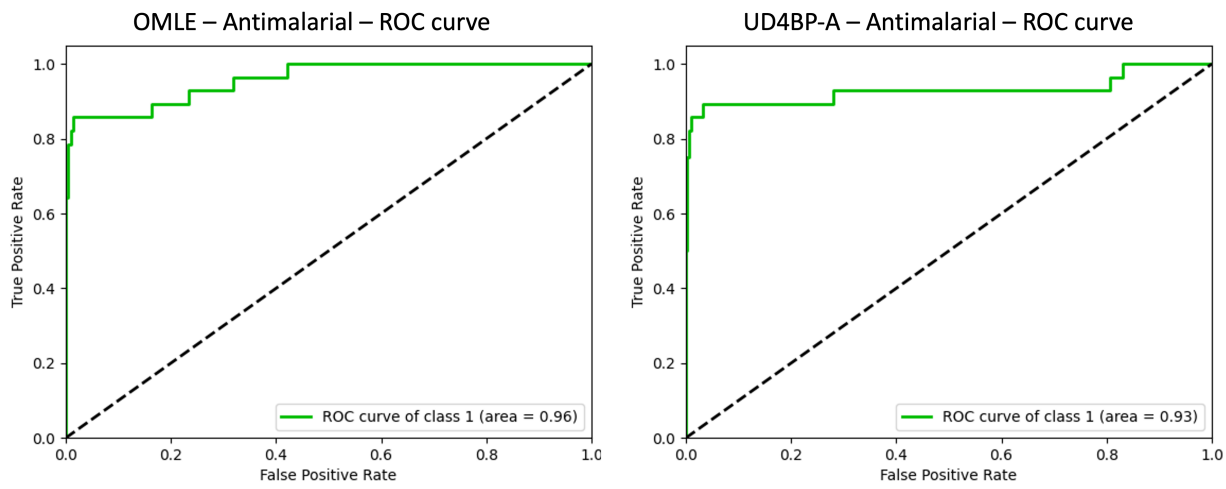

Fig. 23. ROC curves of the original Antimalarial 1 dataset. OMLE (left) and UniDL4BioPep (right).

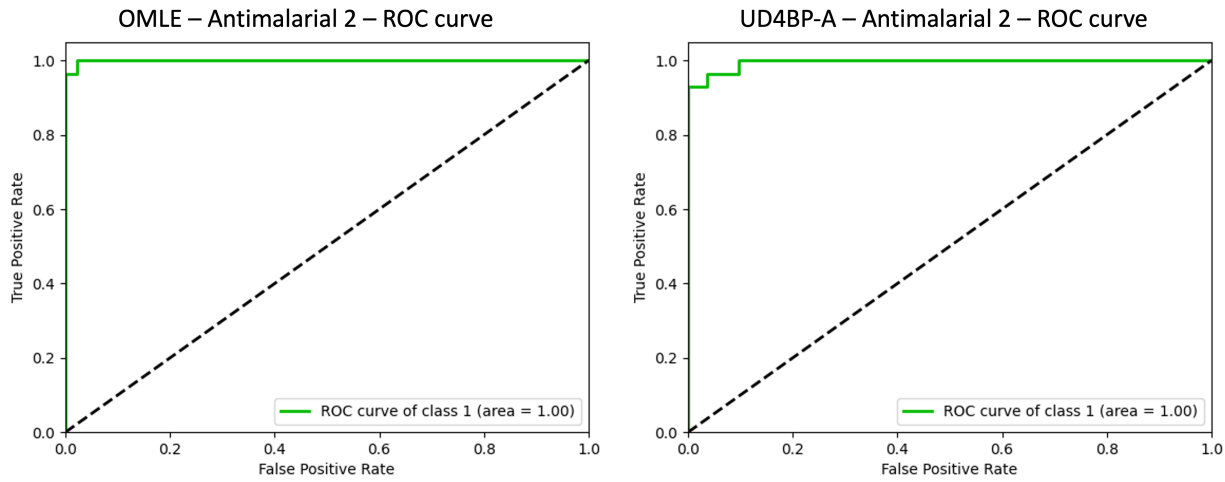

Fig. 24. ROC curves of the original Antimalarial 2 dataset. OMLE (left) and UniDL4BioPep (right).

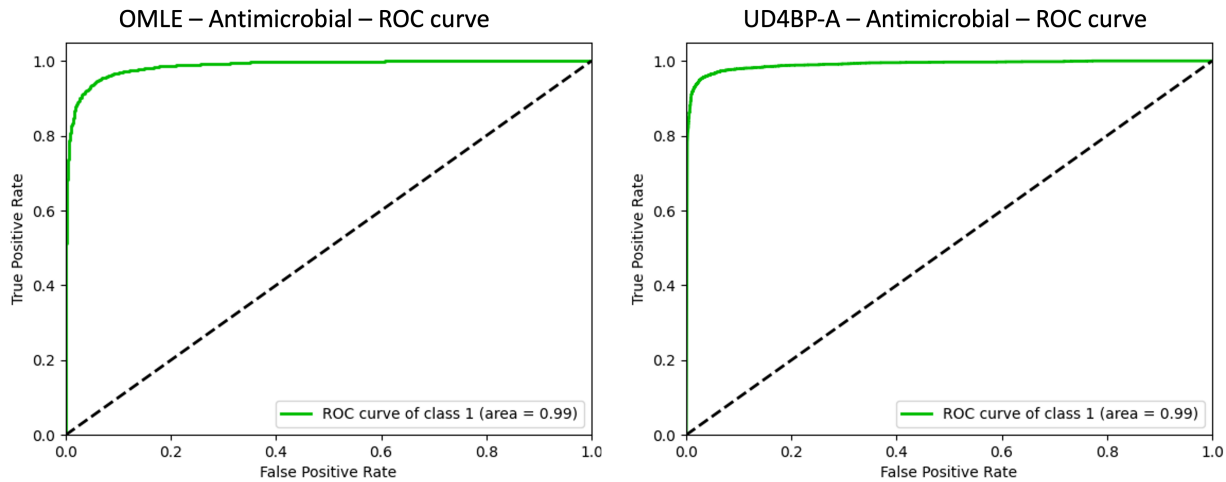

Fig. 25. ROC curves of the original Antimicrobial dataset. OMLE (left) and UniDL4BioPep (right).

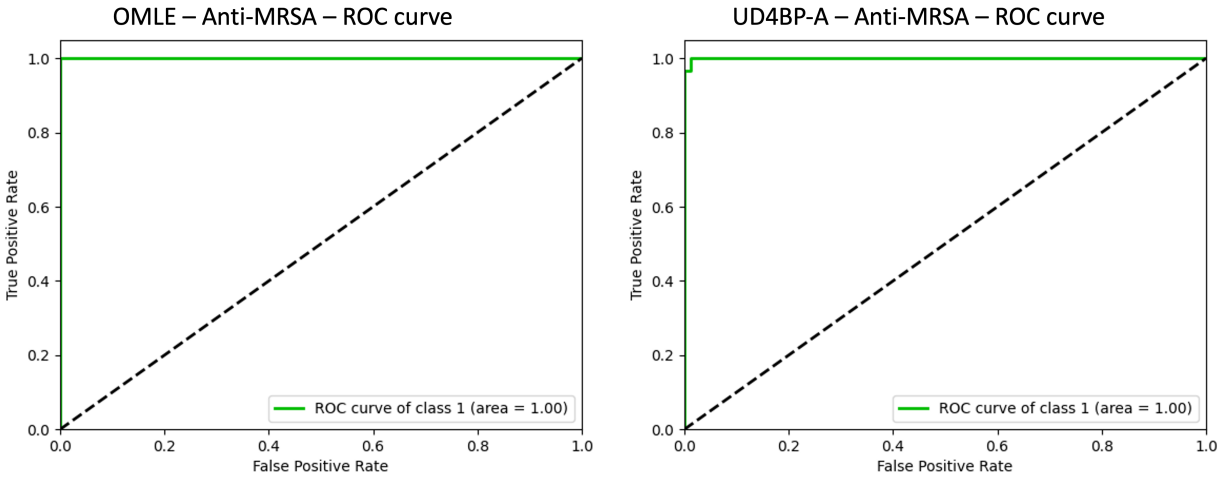

Fig. 26. ROC curves of the original Anti-MRSA dataset. OMLE (left) and UniDL4BioPep (right).

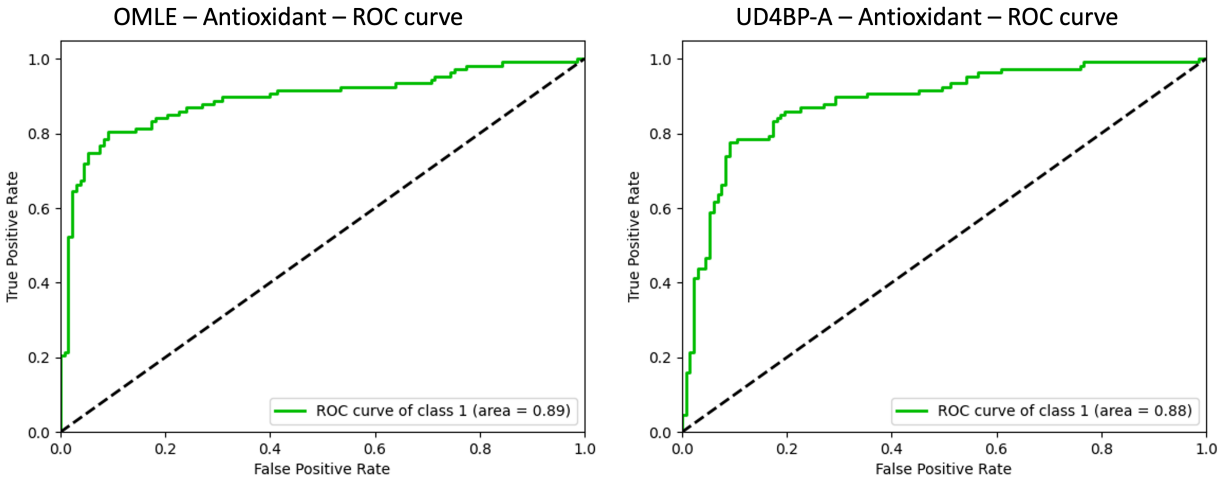

Fig. 27. ROC curves of the original Antioxidant dataset. OMLE (left) and UniDL4BioPep (right).

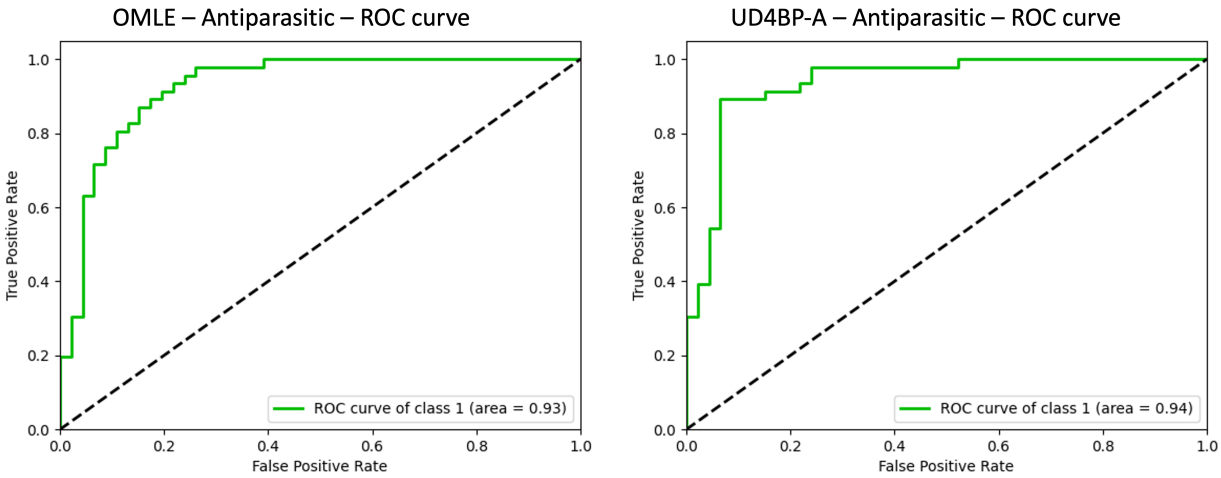

Fig. 28. ROC curves of the original Antiparasitic dataset. OMLE (left) and UniDL4BioPep (right).

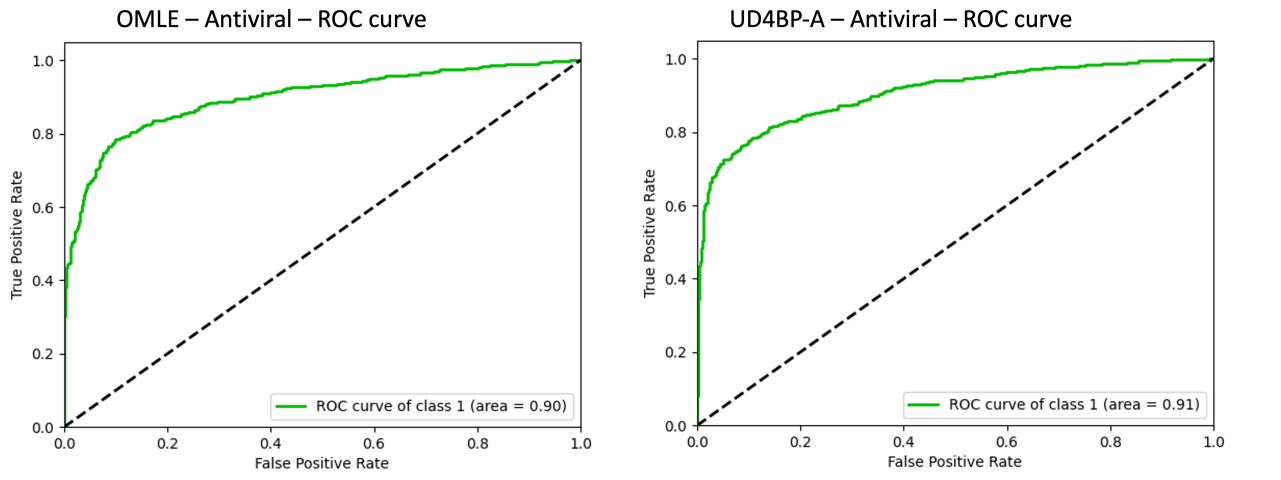

Fig. 29. ROC curves of the original Antiviral dataset. OMLE (left) and UniDL4BioPep (right).

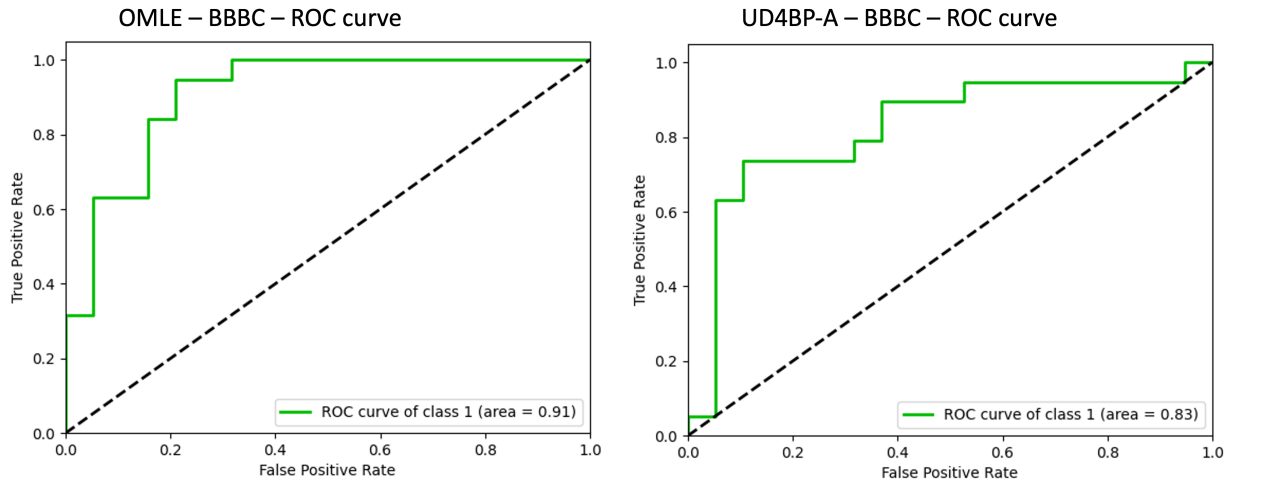

Fig. 30. ROC curves of the original Blood-brain barrier crossing dataset. OMLE (left) and UniDL4BioPep (right).

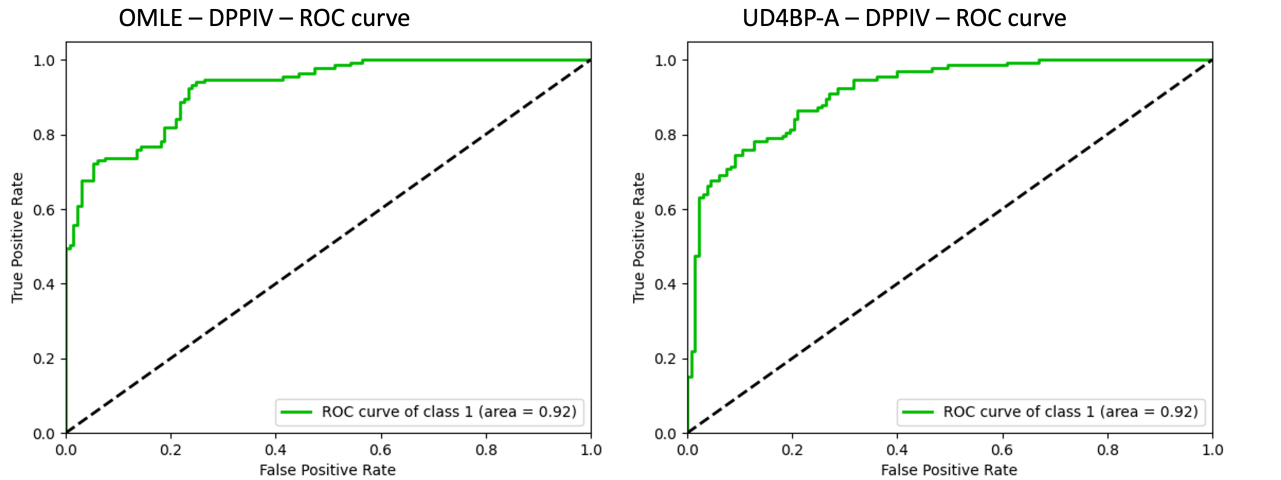

Fig. 31. ROC curves of the original DPPIV inhibitor dataset. OMLE (left) and UniDL4BioPep (right).

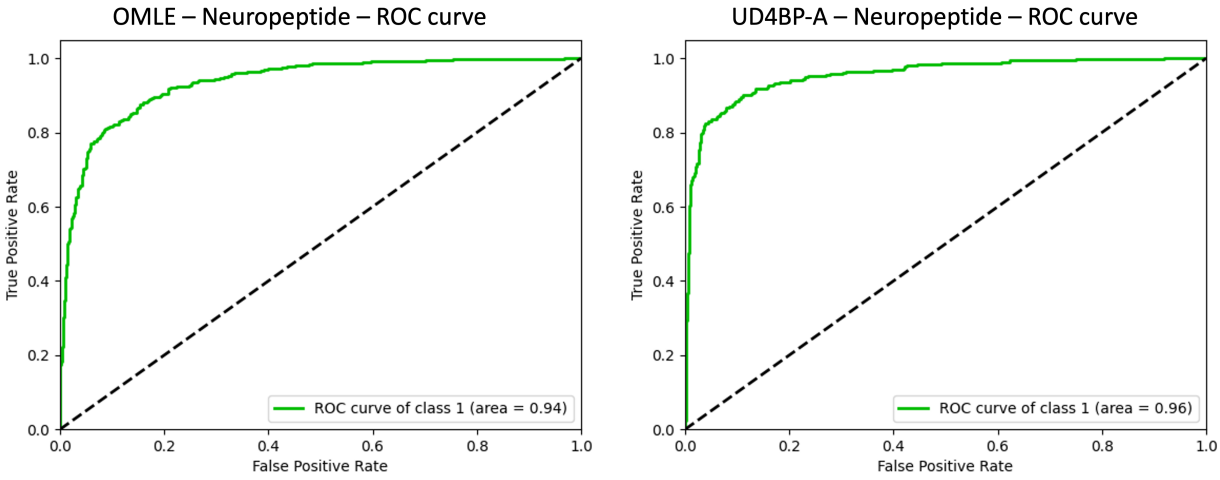

Fig. 32. ROC curves of the original Neuropeptide dataset. OMLE (left) and UniDL4BioPep (right).

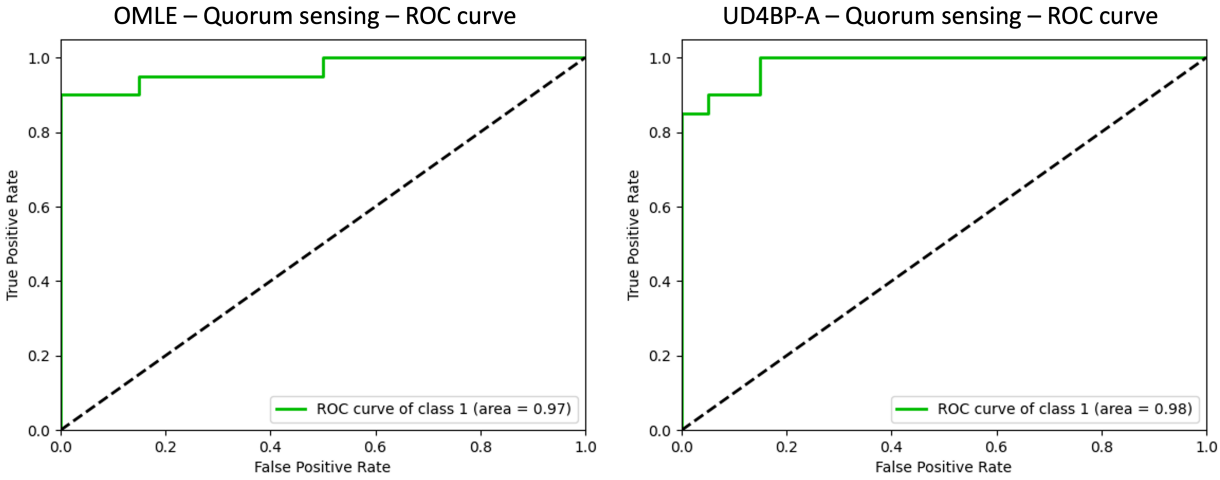

Fig. 33. ROC curves of the original Quorum sensing dataset. OMLE (left) and UniDL4BioPep (right).

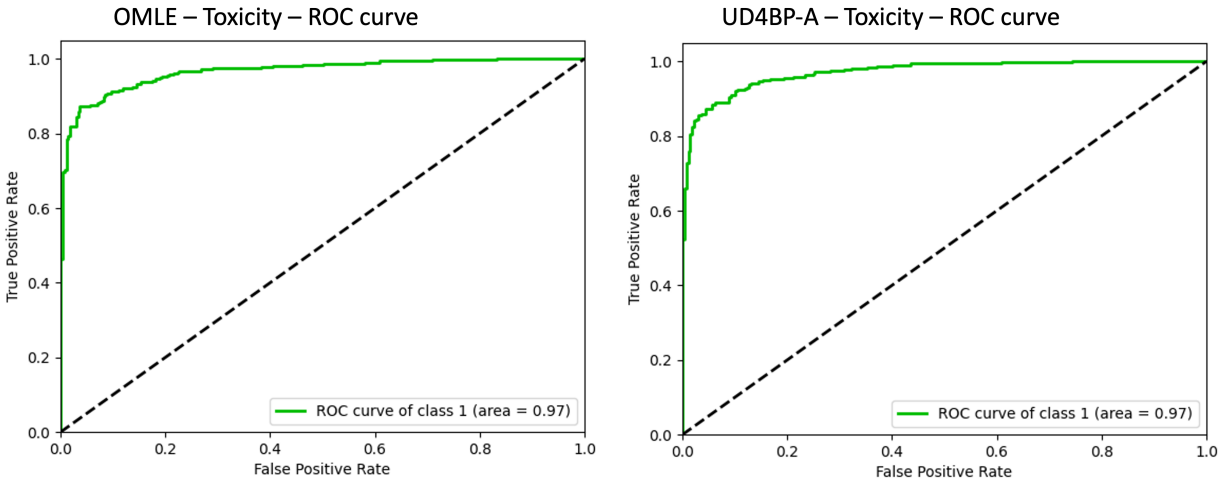

Fig. 34. ROC curves of the original Toxicity dataset. OMLE (left) and UniDL4BioPep (right).

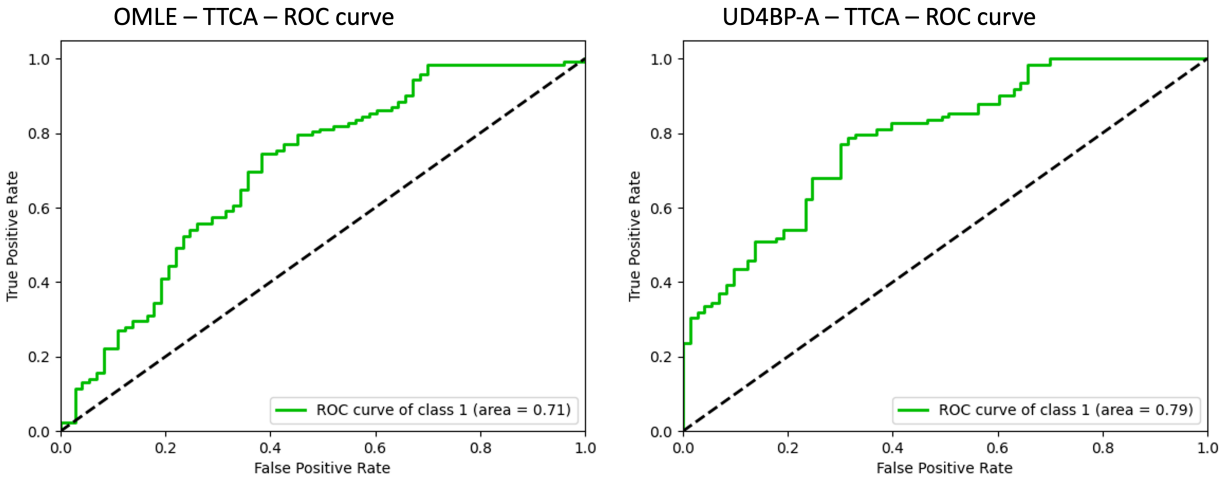

Fig. 35. ROC curves of the original Tumor T-cell antigen dataset. OMLE (left) and UniDL4BioPep (right).

Table 9. Metrics for the evaluation of training strategies in the new datasets. Errors represent the standard error of the mean across three different runs. All values correspond to Matthew's Correlation Coefficient; NegSearch: Dataset with new negative peptides; HP: Homology-based dataset partitioning module; MCC: Mathew's correlation coeficient; BBBC: Brain-blood barrier crossing; TTCA: Tumor T-cell antigens.

Dataset	OMLE NegSearch	UDL4BP-A NegSearch	OMLE NegSearch+HP	UDL4BP NegSearch+HP
Antibacterial	0.663 ± 0.004	0.656 ± 0.004	0.45 ± 0.01	0.42 ± 0.02
ACE inhibitor	0.547 ± 0.004	0.578 ± 0.004	0.57 ± 0.01	0.61 ± 0.03
Anticancer	0.547 ± 0.004	0.64 ± 0.03	0.33 ± 0.01	0.30 ± 0.02
Antifungal	0.78 ± 0.02	0.76 ± 0.02	0.20 ± 0.02	0.26 ± 0.02
Antimalarial	0.51 ± 0.08	0.41 ± 0.08	0.39 ± 0.06	0.33 ± 0.04
Antimicrobial	0.774 ± 0.004	0.749 ± 0.004	0.292 ± 0.009	0.32 ± 0.01
Antioxidant	0.31 ± 0.04	0.34 ± 0.04	0.26 ± 0.06	0.33 ± 0.05
Antiparasitic	0.58 ± 0.07	0.51 ± 0.07	0.41 ± 0.07	0.40 ± 0.06
Antiviral	0.720 ± 0.003	0.725 ± 0.003	0.520 ± 0.009	0.49 ± 0.08
BBBC	0.13 ± 0.09	0.28 ± 0.09	0.08 ± 0.08	0.19 ± 0.06
DPPIV inhibitor	0.494 ± 0.007	0.562 ± 0.007	0.47 ± 0.03	0.56 ± 0.04
Anti-MRSA	0.651 ± 0.01	0.74 ± 0.01	0.41 ± 0.09	0.49 ± 0.03
Neuropeptide	0.701 ± 0.007	0.722 ± 0.006	0.64 ± 0.01	0.708 ± 0.008
Quorum sensing	0.70 ± 0.04	0.72 ± 0.04	0.65 ± 0.08	0.68 ± 0.01
Toxicity	0.65 ± 0.03	0.64 ± 0.03	0.26 ± 0.006	0.39 ± 0.01
TTCA	0.75 ± 0.02	0.77 ± 0.02	0.74 ± 0.02	0.80 ± 0.02

G. AutoPeptideML

Recommendations for using AutoPeptideML and reporting its results

This section explores how the structure of the outputs from AutoPeptideML facilitates compliance with DOME guidelines (Walsh et al., 2021), nevertheless, it is important to note that no system can fully avoid its misuse or abuse and the ultimate responsibility of following proper guidelines and accurately reporting the results lies in the final users.

- **Data:** The algorithm ensures independence between the optimisation (training) and evaluation (test) sets. The hyperparameter optimisation and model selection, which can be considered as meta-optimisation strategies, relies on n-fold cross-validation and maintains the independence of the testing set. Further, the constraints upon the algorithm in the web-server application impedes malpractices like the manual curation of parameters to meta-optimise the results in the independent test sets.

  The datasets generated during the automatic search for negative samples, the train/test partitions, and the n train/validation folds are included in the ZIP-compressed output file, thus making their release and sharing easy. The automatic search for negatives is also compliant with the recommendation that the distribution of the data is representative of the domain in which the model is going to be applied. The use of random seeds for any stochastic process improves the reproducibility when the same exact datasets are used, thus guaranteeing that different runs will produce similar results.
- **Optimisation:** Metrics for each fold in cross-validation are provided alongside the final evaluation metrics of the model so that train versus test error can be calculated as a measure of possible under- or over-fitting. The hyper-parameter configurations of the final models are included in the output file and are therefore easy to share.
- **Model:** PLMs are not directly explainable and it follows that models built on top of their representations are thus not explainable.
- **Evaluation:** Models are evaluated with a wide array of metrics and a PDF summary of the main model performance plots and evaluation metrics is provided with a guide on how to interpret them depending on different application contexts meant for researchers that are not familiar with ML concepts. Most common problems when analysing evaluation metrics arise when working with imbalanced testing datasets, the automatic dataset construction module bypasses this problem by generating balanced datasets.

Output

The output for AutoPeptideML is generated in a ZIP-compressed directory with the following subdirectories:

- `apml_config.json`: File describing the configuration settings used to run AutoPeptideML. It allows the reproduction of experiments as it also contains the seed for the pseudo-random number generator for all stochastic processes.
- `best_configs`: It is a subdirectory containing the best combination of hyperparameters found for all models. It will contain as many configuration files as separate hyperparameter searches run. By default, it will be three.
- `ensemble`: It is a subdirectory that contains the trained models.
- `evaluation_data`: It is a subdirectory that contains two files: 1) `cross-validation.csv`, which is a CSV file with the model performance metrics during hyperparameter optimisation; 2) `test_scores.csv`, which is a CSV file with the final model performance metrics from the evaluation against the testing set.
- `figures`: It is a subdirectory that contains different interesting figures for analysing model performance including: ROC curve, calibration curve, confusion matrix, and precision-recall curve.
- `folds` It is a subdirectory that contains the n cross-validation folds, allows for reproducing the experiments.
- `splits` It is a subdirectory that contains the train and test partitions, allows for reproducing the evaluation.
- `summary.pdf`: It is an automatically generated summary of the evaluation metrics and all the figures in the `figures` subdirectory and guidance on how to interpret both metrics and figures.