Digital Discovery

Accepted Manuscript

This article can be cited before page numbers have been issued, to do this please use: F. T. Achimba, A. Bybordi, M. Gelashvili, J. Ramirez, A. Raja, W. Qiu and M. Holford, *Digital Discovery*, 2026, DOI: 10.1039/D5DD00498E.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/digitaldiscovery

View Article Online
DOI: 10.1039/D5DD00498E

Molecular Arms Race Classifier for Decrypting Venom Peptide and Ion Channel Interactions

Authors: Favour Achimba[1,5#], Arezoo Bybordi[2,6#], Mariam Gelashvili[5,7], Jessy Ramirez[5,8], Anita Raja[2,6], Weigang Qiu[3,7,9] and Mandë Holford [1,3,4,5,10,11*]

[1] The Graduate Center, Program in Biochemistry, City University of New York, NY, USA

[2] The Graduate Center, Program in Computer Science, City University of New York, NY, USA

[3] The Graduate Center, Program in Biology, City University of New York, NY, USA

[4] The Graduate Center, Program in Chemistry, City University of New York, NY, USA

[5] Department of Chemistry, Hunter College, City University of New York, NY, USA

[6] Department of Computer Science, Hunter College, City University of New York, NY, USA

[7] Department of Biological Sciences, Hunter College, City University of New York, NY, USA

[8] Department of Biomedical Engineering, Tandon School of Engineering, New York University, NY, USA

[9] Department of Physiology and Biophysics & Institute for Computational Biomedicine, Weill Cornell Medical College, New York, USA

[10] Invertebrate Zoology, American Museum of Natural History, New York, NY, USA

[11] Department of Organismic and Evolutionary Biology, Harvard University, Cambridge, MA, USA

Corresponding Author Email: mholford@fas.harvard.edu*

# F.A. and A.B contributed equally to this work

Key words: venom peptides, machine learning, structure prediction, ion channels, teretoxins, Random Forest classifier

View Article Online
DOI: 10.1039/D5DD00498E

Abstract

Animal venoms comprise an astonishing number of peptides, proteins and small molecules. The diversity of venom compounds arises from evolutionary adaptations resulting in both offensive and defensive traits in predators and prey alike. This concept, referred to as "arms race", underpins the specificity and selectivity of venom compounds for certain molecular targets, like ion channels. Ion channels are essential regulators of cellular processes, and their dysfunction or dysregulation facilitates a wide range of diseases. Venom peptides and their derivatives are powerful modulators of ion channel activity, with several already in clinical use as FDA-approved therapeutics. Despite the remarkable potential of venom compounds, for the majority, their ion channel targets and modes of action remain largely uncharacterized. We hypothesize that venom peptides are constrained by and converge on molecular structures and targets despite their rapid sequence divergence due to arms race evolution. Here, we introduce a machine learning approach termed Molecular Arms Race Classifier (MARC), which predicts the ion channel targets – sodium, potassium, and calcium ion channels – of cysteine-rich venom peptides. MARC leverages evolutionary scale modeling (ESM) for feature extraction along with random forest classification to enable predictive functional annotation of venom compounds by their putative ion channel targets. MARC performs multi-class classification across four categories (sodium, calcium, potassium and non-ion-channels), to predict the ion channel targets of novel cysteine-rich venom compounds. We trained, tested, and cross-validated MARC on 5,165 peptide and protein sequences sourced from in-house venom gland transcriptomes and public databases spanning diverse taxa, including sea anemones, snakes, scorpions, spiders, cone snails, and terebrid snails. We identified 28 novel terebrid snail venom peptides (teretoxins) predicted to target potassium ion channels. Orthogonal validation using docking and molecular dynamics simulations suggests the stability of the best docked pose of the K^+ channels, KcsA and MthK, to the teretoxin, Cje1.9, supporting MARC's robust prediction of Cje1.9 as a K^+ channel-targeting peptide. Taken together, these results indicate that MARC is a cost-effective method for screening vast peptide and protein libraries to identify potential ion channel targeting compounds, paving the way to design novel peptides selectively targeting distinct ion channel classes.

View Article Online
DOI: 10.1039/D5DD00498E

Introduction

The repetitious tale of a venomous snake biting a mongoose and the mongoose resisting the negative effects of the deadly bite illustrates the evolutionary adaptations that support a current theory of how predator-prey interactions have led to an enormous diversity of venom compounds [1]. Such "arms race" interactions promote offensive and defensive traits in predator and prey that bolster the specificity and selectivity of venom compounds for certain molecular targets like ion channels[1]. The interaction of venom peptides with their respective ion channel molecular targets is shaped by over millions of years of evolution and substantiated by the arms-race concept wherein venom compounds and ion channels are constantly evolving to support the development of new offensive or defensive traits in predator-prey interactions like between the snake and the mongoose[1,2]. Despite the rapid co-diversification of the primary amino-acid sequences of venom peptides and their target channels driven by the arms-race co-evolution, these molecules are constrained by structural integrity and functional bioactivities[3]. Consequently, we hypothesize that venom peptides are constrained by and converge to a limited set of structural features, each set of which corresponds to a distinct class of molecular targets of ion channels.

Many diseases are described as "channelopathies" i.e. diseases resulting from ion channel dysfunction or dysregulation[4,5]. Identifying ligands that can selectively modulate dysregulated ion channels is a key strategy in drug development for such diseases[6,7]. Several existing drugs that target ion channels are peptides or peptide derivatives from the venom of animals, like ziconotide, from the venom of *Conus magus*, which targets N-type calcium (Ca^{2+}) channels and is approved for the treatment of chronic pain in HIV and cancer patients[8]. Ozempic® from the Gila monster modulates the Glucagon-like peptide-1 (GLP-1) receptor and is approved for the treatment of diabetes and obesity[9]. Venom peptides have also induced apoptosis and tumor regression by targeting dysregulated ion channels in various cancer types [7],[9],[10]. Importantly, venom peptides can modulate ligand and voltage gated ion channels and receptors on the cell membrane, or act as pore blockers interfering with ion channel influx by gating the pore of the ion channel. For instance, tetrodotoxin[10], a common voltage-gated sodium channel (Na_v) blocking neurotoxin acts by modulating Na^+ conductance, whereas cone snail μ-conotoxin peptides[11,12] act by blocking the pore of Na_v channels. Despite the promise of venom compounds, the magnitude of uncharacterized venom peptides is staggering, presenting a daunting task for accurately determining their ion channel targets.

View Article Online
DOI: 10.1039/D5DD00498E

Venomous animals are estimated to be about ~15-30% of all animal biodiversity[13–15]. Cone snail venom peptides alone represent a class of approximately 10,000 distinct peptide sequences deposited in ConoServer, of which only a small fraction have annotated functions[16]. A UniProt search of "scorpion toxin" on 2/4/2025 reveals 13,267 scorpion toxin sequence entries with only 1,163 reviewed and 810 (~ 6%) confirmed to have protein level evidence [17]. A search of "scorpion venom peptide" on UniProt (search date: 2/4/2025) reveals 1037 results with 522 reviewed and 368 (~35%) with protein level evidence[17]. While there is much work to be done to identify undiscovered venom peptides, deorphanizing the peptides already in data repositories would generate a large library of potentially pharmaceutically active hits **(Figure 1)**. A computational approach that can *a priori* decrypt the ion channel target of venom peptides enables large-scale screening preceding costly synthesis and bioactivity assays.

Here we introduce a method to address this need, based on an "arms-race" model. Molecular Arms Race Classifier (MARC) is a novel machine learning (ML) based approach which leverages evolutionary scale modeling (ESM) for feature extraction in tandem with random forest classification to enable predictive functional annotation of venom compounds by their putative ion channel targets **(Figure 1)**. MARC utilizes machine learning to predict the potential interactions between venom peptides and ion channels, providing a cost-effective and data-driven approach to characterizing venom peptide ligands.

There is an increasing trend to use *in silico* methods to mine venom peptide libraries[18–21]. Machine learning methods have been applied to various biological applications, including proteome analysis, which involves determining the sequence, location, and function of protein-encoding genes[22]. Deep learning models, a subset of machine learning, have demonstrated significant success in genome engineering [23]. With regards to ion channel classification, several algorithms have demonstrated the potential and challenges with *in silico* methods. For example, Support Vector Machines (SVM), Hidden Markov Models (HMM), and k-nearest-neighbors (k-NN) have been widely used for classifying ion channel targeting peptides based on features such as amino acid composition, dipeptide composition, and physiochemical properties [24,25]. While these methods achieved high accuracy, they primarily focus on overall classification performance without addressing potential false positives in multi-class settings, for example classifying peptides targeting four ion channels [24]. k-NN-based approaches have been applied to

View Article Online
DOI: 10.1039/D5DD00498E

classify peptides targeting potassium channels but lack the ability to generalize to peptides targeting other ion channel types, such as sodium or calcium channels [25]. More recently, machine learning models incorporating feature extraction techniques, such as k-skip-n-gram, have been used to classify peptides targeting voltage-gated potassium, calcium, sodium, and anion channels, employing SVM and Random Forest [26]. IonchanPred 2.0 combines peptide composition with other physiochemical properties and uses SVM as a machine learning model to predict ion channels [27]. The ESM model and its versions, such as ESMFold (for predicting protein structure) and ESM-2, provide embeddings that can be used for various downstream tasks[28]. ESMFold has also been used to model ion channel structures [29]. While these existing algorithms are promising, there is still space to explore a comprehensive approach that can analyze venom peptides from multiple species and various ion channel classes.

Our MARC model performs multi-class classification across four categories – Ca^{2+}, Na^{+}, K^{+} ion-channel targeting compounds, and non-ion-channel targeting compounds – and is trained to predict the most likely ion channel interactions for venom compounds across the aforementioned categories (**Figure 1)**. MARC's predictions were benchmarked based on training with ion channel targeting venom peptide sequences from validated databases that include venom peptides that activate, inactivate, or block their respective channels.

Evaluation of MARC's predictions considers accuracy and multiple metrics (recall, precision and F1-score) to ensure the model's robust performance across voltage- and ligand-gated ion channel classes. Specifically, we tested MARC's prediction abilities against a large dataset of novel terebrid snail peptides, teretoxins. Teretoxins are an untapped resource of marine snail venom peptides that are significantly different from conotoxins and have yet to be characterized for ion channel activity[30,31]. AlphaFold[32] 2.0 and Schrodinger Maestro[33] were used to visualize potential molecular interactions between selected teretoxin peptides and the respective ion channel target to validate the accuracy of MARC's predictions.

View Article Online
DOI: 10.1039/D5DD00498E

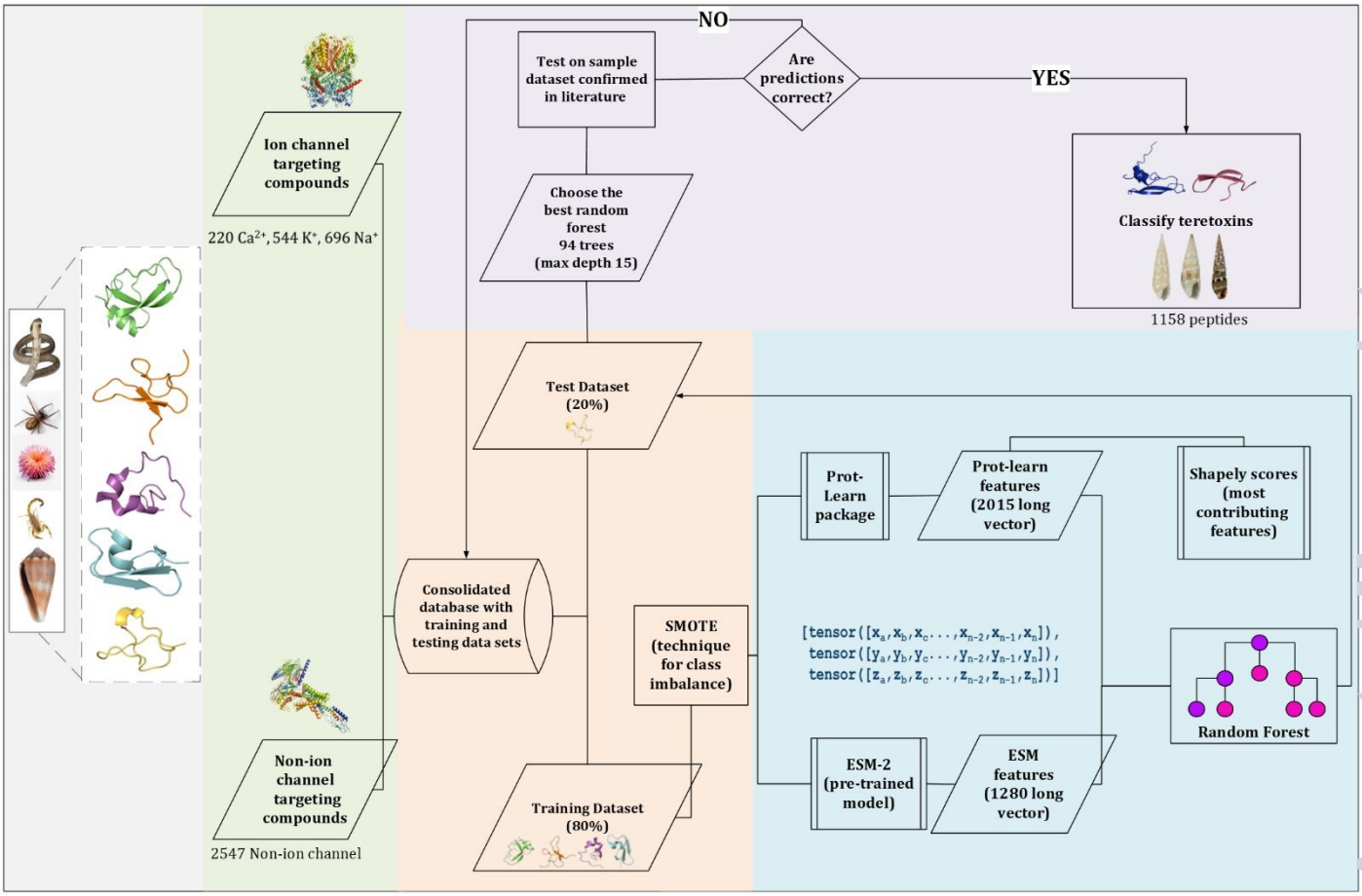


Figure 1. Overall pipeline for Molecular Arms Race Classifier (MARC) training, testing and teretoxin venom peptide classification. *From left to right*: Ion channel (1460) and non-channel (2547) targeting venom peptides from snakes, scorpions, spiders, sea anemones, scorpions, and cone snails were used to generate the overall dataset (grey and green shaded images), which was split into training (80%) and testing (20%) sets (peach shaded images). The training set was assessed for class imbalance and SMOTE was employed to address any imbalances (peach shaded images). ESM and Prot-Learn were used to generate features to be further used as input to generate random forests (blue shaded images). Following this, Random Forests were generated, and the test data set was used to select the best random forest (blue shaded images). In the case of Prot-Learn features, Shapley values were used to interpret the model's predictions. Shapley values measure the contribution of each feature to the prediction. The model was then tested on a separate randomly selected sample dataset with known ion channel targets to further assess prediction accuracy (purple shaded images). If the predictions are not accurate, the training and test datasets

View Article Online
DOI: 10.1039/D5DD00498E

are manually reviewed and the entire training process repeated. If the predictions are accurate, the model is applied to classify the novel teretoxin dataset consisting of 1158 peptides (purple shaded images).

Materials and Methods

Teretoxin sample collection, preparation, RNA preparation, and transcriptome sequencing

A total of 25 mollusk samples were collected from three distinct geographic locations: Mozambique (Inhaca, 2011) and Papua New Guinea (Kavieng, 2014; Madang, 2015), with the Muséum National d'Histoire Naturelle (MNHN)[34,35]. Of these, 24 samples are from the Terebridae family, while one sample (*Iotyrris cingulifera*) belongs to the Turridae family. The collection of samples is a mixture of venom gland (n=24) and foot tissue (n=1) transcriptomes. Raw reads were deposited in the Sequence Read Archive (SRA) linked here (https://dataview.ncbi.nlm.nih.gov/object/PRJNA1222918?reviewer=pjds78dfcg03jp06r165chm5f). **Supplementary Table 1** details the in-house transcriptomes used in this study.

Transcriptome de novo assembly for teretoxins

The raw sequencing reads obtained from the Illumina HiSeq platform for the 25 mollusk samples were first subjected to trimming and adapter removal using Trimmomatic[36] (v0.39). The parameters used for Trimmomatic removed any leading bases with a quality score below 3, removed any trailing bases with a quality score below 3, scanned the read with a 4-base-wide sliding window and removed bases if the average quality per base dropped below 15, and discarded reads with 36 bases or less after trimming. Subsequently, the quality of the processed reads was assessed using FastQC (v0.11.6)[37]. Since there is no reference genome for terebrids, these high-quality reads were assembled *de novo* with Trinity (v2.5.1) as previously described[38,39]. The assembled transcriptomes were evaluated for their quality and completeness. Using TrinityStats, number of transcripts, total genes, percent GC, and N50 were extracted. The transcriptomes were evaluated for completeness using BUSCO (v5.4.2)[40,41]. Metazoa was selected as the lineage dataset and the "genome mode" parameter was utilized. All pre-processing, quality control, and transcriptome assembly were performed using the in-house

View Article Online
DOI: 10.1039/D5DD00498E

computing cluster that features an x86_64 architecture, supports both 32-bit and 64-bit CPU operation modes, is equipped with 24 CPUs, and 124GB RAM.

Venom peptide identification

Using a bioinformatic pipeline previously applied in the lab[34,35], open reading frames (ORFs) were predicted using TransDecoder (v5.5.0)[42], a bioinformatics tool optimized for transcripts assembled by Trinity. The resulting ORF FASTA files were queried against our in-house teretoxin database using BLAST (v2.13.0)[43]. BLAST hits were filtered to retain sequences that initiated with a methionine (M) residue, exhibited a minimum sequence identity of 90%, and passed thresholds for both e-value and bitscore significance. Duplicate venom peptide precursor sequences were filtered out within, but not between, each species using an exact string match search. Using this pipeline, a total of 1,158 putative venom peptides were identified across the assembled transcriptomes.

Addressing class imbalance using SMOTE

Class imbalance occurs when certain classes in a dataset are underrepresented compared to others, which can lead to biased models. To address class imbalance in the dataset, we employed the Synthetic Minority Over-sampling Technique (SMOTE)[44]. In our dataset, the total sample/venom compound counts for each class are as follows: Sodium (Na^+): 696, Potassium (K^+): 544, Calcium (Ca^{2+}): 220, and Non-Ion Channel Targeting: 2547. The number of non-ion channel targeting samples were significantly larger than Ca^{2+} targeting samples (and other minority classes). This imbalance could have led to the development of a bias in the model, erroneously predicting a higher probability for random venom compounds to belong to the overrepresented class – in this case, Non-Ion Channel Targeting. This could have led to the misclassification of venom compounds that target Ca^{2+} channels (or other underrepresented ion channels) as non-ion channel targeting compounds, which may hinder performance and reduce the model's generalizability. SMOTE generates new samples for underrepresented classes, ensuring balanced sample sizes across all classes. In our implementation, the synthetic samples were generated directly from the embedding features (ESM or ProtLearn), rather than from raw

View Article Online
DOI: 10.1039/D5DD00498E

sequences, ensuring consistency with the input representation used for classification. This approach was particularly effective in augmenting the Ca^{2+} channel class in our dataset. Applying SMOTE resulted in equalizing the number of samples in all four classes. This was especially beneficial for the Ca^{2+} channel class, as it had the fewest samples prior to augmentation. By generating synthetic samples for this underrepresented class, SMOTE ensured a more balanced distribution of data. The specific point in the pipeline where this method is applied is described in **Figure 1**. The imbalance in our dataset (approximately 2:1 non-ion channel to ion channel) reflects the biological reality of venom, where non-ion channel targeting peptides significantly outnumber ion channel-specific ones. We prioritized maximizing data utilization over undersampling, as discarding valid non-ion channel sequences would artificially reduce the model's exposure to the background diversity of venom and likely increase false positives. We employed SMOTE specifically to correct for this statistical imbalance without sacrificing valuable training information from the majority class. This methodological choice is supported by recent computational studies demonstrating that hybrid sampling techniques like SMOTE consistently outperform undersampling in preserving model accuracy on imbalanced biological datasets[45]. Furthermore, our pilot evaluations indicated that training exclusively on the ion-channel targeting subset diminished generalization performance due to the substantially smaller dataset size, which would fundamentally compromise the model's utility as a *de novo* screening tool for unannotated sequence libraries[46,47].

Feature extraction for protein classification

Feature extraction was performed using two complementary approaches. First, we used the Evolutionary Scale Modeling (ESM) deep learning model[28], based on the RoBERTa architecture, a robust variant of BERT (Bidirectional Encoder Representations from Transformers)[48]. ESM is a protein language model trained on protein sequences, producing embeddings – vectors of length 1280 – that capture the underlying biological properties of amino acid sequences. In our study, we specifically used the pretrained **ESM-2 model** (esm.pretrained.esm2_t33_650M_UR50D). This choice was motivated by its demonstrated ability to capture context-dependent sequence features in protein data. These embeddings serve as compact numerical representations of the hidden information encoded in protein sequences, which are crucial for downstream machine learning tasks. We selected ESM-2 (esm2_t33_650M_UR50D) because it has demonstrated state-of-the-

View Article Online
DOI: 10.1039/D5DD00498E

art performance on protein classification tasks, is openly available, and has been extensively benchmarked across diverse biological sequence datasets. These factors made it a robust and transparent choice for our study. While alternative embeddings such as ProtT5[49] and Ankh[50] are promising, systematic benchmarking across all models was beyond the scope of this work. The model is built on the transformer architecture, which consists of two key components: an encoder and a decoder. The encoder processes the input sequence, transforming it into a high-dimensional vector that encapsulates its features. The decoder then utilizes this encoded representation for specific downstream tasks, such as predicting protein functions or structures. This feature-rich encoding enables the model to capture intricate patterns in the protein sequences, which are essential for accurate prediction and analysis.

The second approach involved the use of Prot-learn,[51] a Python package designed to extract various biological properties of proteins, such as amino acid indices, composition, hydrophobicity, and other features. From Prot-learn, we extracted a total of 2015 features, which included the length of the protein (1 feature), amino acid composition (20 features), amino acid indices (553 features), N-gram properties (400 features), entropy (1 feature), and advanced sequence descriptors such as ATC (5 features), CKSAAP (400 features), CTD (343 features), CTDC (39 features), CTDT (39 features), CTDD (195 features), Moreau Broto (8 features), and Moran (8 features). These features captured the structural and biological characteristics of proteins comprehensively for classification.

To provide a complementary benchmark, we pursued two independent strategies rather than merging embeddings with handcrafted descriptors: (a) an ESM+Random Forest model leveraging contextual embeddings, and (b) a ProtLearn-based model using explicit biochemical and sequence-order descriptors. This dual approach allowed us to evaluate whether pretrained embeddings capture predictive signals beyond explicit biochemical features, while also preserving interpretability.

Machine learning and feature relevance analysis

A Random Forest classifier[52], a robust machine learning model widely used for classification tasks, was employed to classify ion channels into four categories: Potassium (K^+), Calcium

View Article Online
DOI: 10.1039/D5DD00498E

(Ca^{2+}), Sodium (Na^{+}), and Non-Ion Channel Targeting. We selected Random Forest and XGBoost because both methods have demonstrated strong performance in prior biological prediction tasks, including peptide and protein classification. They are well-suited to high-dimensional feature spaces, handle class imbalance effectively, and provide interpretable outputs. In preliminary experiments, their performance was sufficient for our study's objectives, so we focused on these two algorithms rather than expanding to additional models. The model constructs an ensemble of decision trees, each predicting the most probable class based on the input features and determines the final classification through majority voting across all trees (**Figure 1**). This approach ensures both accuracy and robustness in distinguishing between different classes.

To optimize the model's performance, we fine-tuned two key hyperparameters: the number of trees in the forest (ranging from 10 to 200, in increments of 10) and the maximum depth of each tree (explored at depths of 5, 7, 8, 10, 15, 20, 25, 30, and 40). We employed a grid search strategy for hyperparameter optimization. While more advanced approaches such as randomized search or Bayesian optimization[53] exist, grid search remains a reliable and widely used method due to its simplicity, transparency, and reproducibility. In our case, it provided effective tuning and satisfactory performance, and we therefore did not pursue more complex optimization strategies. Model performance was evaluated using accuracy, precision, recall, and F1 score, ensuring a comprehensive assessment of classification quality. In addition, Receiver Operating Characteristic (ROC) analysis was used to evaluate classification performance across the four ion channel classes (Na^{+}, K^{+}, Ca^{2+}, and Non-Ion Channel). ROC curves were computed using a one-vs-rest scheme, where the true positive rate (TPR) and false positive rate (FPR) for each class were defined as: $TPR_c = \frac{TP_c}{TP_c+FN_c}, FPR_c = \frac{FP_c}{FP_c+TN_c}$.

We also report micro-averaged ROC curves, where predictions across all classes are pooled. The micro-averaged true positive rate and false positive rate were defined as:

$$Micro\ TPR = \frac{\Sigma_c TP_c}{\Sigma_c(TP_c+FN_c)}, Micro\ FPR = \frac{\Sigma_c FP_c}{\Sigma_c(FP_c+TN_c)}.$$

This ensured that both per-class and overall performance were captured, while accounting for class imbalance. This method assigns different weights to each class based on its sample size, making micro averaging more reflective of overall model performance when classes are imbalanced. In

View Article Online
DOI: 10.1039/D5DD00498E

contrast, the usual average gives equal weight to each class regardless of size, which may be less informative in the presence of class imbalance. All performance metrics, including ROC curve analysis, were computed on the held-out 20% test set from the 80:20 stratified split.

We also trained an XGBoost classifier [54,55], a popular ensemble learning model. This model operates by constructing an initial decision tree and improving its accuracy through successive iterations. For the training of the XGBoost classifier, we exclusively utilized ESM features because earlier evaluations using Random Forest revealed that ESM-based models consistently outperformed those using ProtLearn features. Given the superior performance of ESM features in capturing biologically meaningful patterns, we prioritized them in subsequent models.

Finally, we utilized Shapley values[56] to identify the most significant features contributing to the classification process. Shapley values calculate a feature's contribution by considering all possible combinations of features, ensuring that each feature receives an accurate share of the credit for the prediction regardless of interactions or dependencies between them. This analysis confirmed that specific physicochemical properties, such as distinct amino acid indices, were the primary drivers of the ProtLearn model's predictions, grounding the model in biological reality **(for more detail, please see Supplementary Figure 9)**. This analysis was conducted exclusively on the Prot-learn features due to their interpretability, enabling us to determine which biological properties were most influential in classifying the target ion channels. All analysis involving MARC were enabled by the Bridges-2 cluster at the Pittsburgh Supercomputing Center[57,58].

Structure prediction

Venom peptide structure prediction was performed using AlphaFold2[32] on a Dell GPU computer equipped with an NVIDIA GPU, 5 TB of storage, and 32 GB of RAM, provided by MH. AlphaFold2 (AF2) was cloned onto the GPU system following the steps outlined on the AlphaFold GitHub page maintained by Google DeepMind (https://github.com/google-deepmind/alphafold). Prior to the installation of AF2, Docker, the NVIDIA Container Toolkit, and the required genetic databases (BFD, MGnify, PDB70, PDB, PDB seqres, UniRef30, UniProt, and UniRef90) were installed. The peptides were predicted using existing structural templates in eight publicly available databases above. Additionally, a multiple sequence alignment(MSA) of an in-house

View Article Online
DOI: 10.1039/D5DD00498E

database of teretoxins was included with the MSA from the default databases to improve teretoxin structure prediction. For conotoxins and other venom peptides, predictions were based solely on the MSAs from the templates available in the eight databases listed above. The command-line script provided on the AlphaFold GitHub page was modified to accommodate the specific requirements of our study. The best models with the highest pLDDT score were selected for downstream analysis for each predicted peptide.

Structure clustering and multiple sequence alignment

Clustering of predicted teretoxin structures was performed using qTMclust from US-align which categorizes similar structures using the template modeling (TM) score and root mean square deviation (RMSD) values. A threshold of 0.7 was selected for clustering because at this threshold cluster sizes were robust and only included closely related members with similar disulfide connectivities. Multiple sequence alignment (MSA) of teretoxins complete sequence (signal + pro + mature peptide) was performed with MAFFT v7.505 using the E-INS-i strategy which is suitable for sequences with long unalignable regions. The BLOSUM62 model and gap penalties of 1.53, 0.00 and 0.00 were applied. The following command line was used to perform the MSA: `mafft --genafpair --maxiterate 1000 input.fasta > output.fasta.`

Molecular docking and molecular dynamics simulation

All structures were prepared using the Protein Preparation Wizard[59] in Maestro. The preprocessing steps were consistent for all proteins and involved capping termini, filling in missing side chains with Prime, assigning bond orders, replacing hydrogens, and creating zero-order bonds to metals and disulfide bonds. Additionally, heteroatom states were generated using Epik at pH 7.4 with a maximum of one state per structure. The options for optimizing hydrogen bond assignments, minimizing energy, and deleting water molecules were left at default settings.

Molecular docking was performed using Schrödinger's Maestro[33] protein-protein docking feature. Protein-protein docking enabled by the PIPER[60] algorithm was carried out after structure preparation on Maestro. Peptides were assigned unique chain names to facilitate the identification of intermolecular interactions during post-docking analysis. To enhance

View Article Online
DOI: 10.1039/D5DD00498E

meaningful interactions and generate physiologically relevant poses, constraints were applied during docking. The transmembrane and cytoplasmic domains of the ion channels were assigned repulsion constraints, while the extracellular domains were assigned attraction constraints. Default settings were used for docking, generating 70,000 poses, from which the top 30 poses were returned as output. The best poses were selected based on hierarchical classification provided by the software, the number of hydrogen bonds present, and the placement of the ligand (peptide) within the binding site.

The selected poses were prepared for lipid membrane placement using System Builder. Structures of the ion channels of interest were downloaded from the OPM database, which included their orientation in a lipid membrane. During membrane building with Maestro's System Builder, annotated transmembrane, cytoplasmic, and extracellular domains were specified. For structures obtained from the OPM database membrane placement on the docked pose was based on the pre-aligned structure after which charges were recalculated and 0.15 M NaCl added to the system after which a membrane was built using the System Builder option. The selected docked pose was solvated using an SPC solvent model and OPLS4[61] forcefield in an orthorhombic boundary box with distances of 10 angstroms for a, b and c respectively before being subjected to molecular dynamics (MD) simulation using Desmond[62] in Maestro.

Molecular dynamics (MD) simulations were conducted for 10 ns and 50 ns with energy set to 10 and a recording interval of 10 ps, generating approximately 1,000 frames per simulation. The number or particles(constant), pressure(constant) and temperature(constant) (NPT) and number of particles, pressure, surface tension, and temperature (NPγT) ensemble classes were applied for MD simulations with MthK (PDB ID: 4HYO) and KcsA (PDB ID: 2QTO) potassium channels respectively. NPT was also applied for MD simulation of the experimental complex of KcsA to charybdotoxin (PDB ID: 2A9H). Simulation interaction analysis was performed using the Simulation Interaction Diagram tool in Maestro. Average RMSDs and average interaction count were calculated as the mean of RMSD or interaction counts per frame over the number of frames.

Results and Discussion

View Article Online
DOI: 10.1039/D5DD00498E

Molecular Arms Race Classifier (MARC) testing and training datasets generated using a diverse library of venom peptides and proteins

For the testing and training of the MARC model, we examined 4,007 venom compounds, not including teretoxins, from over 500 species of venomous snakes, scorpions, spiders, cone snails, and sea anemones. Ion channel targeting and non-ion channel targeting venom compounds from the Molluscan family Conidae (cone snails) and orders Actiniaria (sea anemones), Araneae (spiders), Serpentes (snakes), and Scorpiones (scorpions) were downloaded from UniProt on 11/30/2023 and 01/28/2024, respectively. Search criteria were limited to reviewed entries (SwissProt) with non-fragmented sequences. Entries with ambiguous amino acid calls ("X", "B", "J", "Z") were filtered out of the training and test dataset. Ion channel targeting venom compounds were pulled from SwissProt using the keywords "toxin" (KW-0800) and "ion channel impairing toxin" (KW-0872). Non-ion channel targeting venom compounds were limited to compounds that contained the keyword "toxin" but not the keyword "ion channeling impairing toxin". The ion channel targeting compounds from cone snails were taken from Zhang et al. and filtered using similar criteria, the exception being that fragmented sequences were retained[44]. The molecular targets of each venom compound were generalized into four categories: Ca^{2+} channel, K^{+} channel, Na^{+} channel, and non-ion channel. **Table 1** summarizes the composition of the consolidated dataset, and **Supplementary Tables 2 and 3** detail the dataset in full. The extracted features from the dataset include signal sequence, peptide or protein sequence, and cysteine framework using customized Python scripts.

The multiple species and diversity of channel classes compiled in MARC's consolidated training and testing sets are unlike prior databases used in other venom compound classifier models previously reported[26]. The variety of organisms and molecular targets were meant to emulate the "arms race" as venom peptides and proteins evolve over time. The non–ion-channel class includes a diverse range of venom-derived components, including enzymes such as metalloproteinases, serine proteases, three-finger toxins, nicotinic receptors and phospholipases. This is a broad and heterogeneous category incorporating proteins, peptides, and growth factors from the venomous organisms represented in our dataset, with snake, scorpion, and spider venoms comprising the largest contributions. Because this class encompasses multiple distinct modes of bioactivity rather than a single mechanistic category, it is expected that a large number

View Article Online
DOI: 10.1039/D5DD00498E

of peptides in the training dataset fall within this group. Grouping these peptides into a single non–ion-channel class allows MARC to learn features that distinguish ion-channel–targeting peptides from those with other biological activities, thereby improving model generalization rather than overfitting to narrowly defined functional subclasses.

The consolidated dataset of ion channel targeting venom compounds and non-ion channel targeting venom compounds is divided into a training (80%) and a testing (20%) set. We used stratified sampling to split the dataset into training and test sets with an 80:20 ratio. Stratification was applied to ensure that the distribution of both species and ion channel classes remained consistent across the two subsets. To ensure MARC generalizes across the widest chemical space, we used stratified sampling rather than leave-class-out validation. Including all available taxa prevents the loss of lineage-specific structural nuances, maximizing the model's accuracy and robustness for *de novo* discovery. MARC is benchmarked on a separate randomly selected sample dataset with ion channel activity confirmed from literature and if the predictions are correct, we move forward with the classification of the teretoxin venom peptides. If the MARC predictions are incorrect, the consolidated dataset is checked for discrepancies in the sequences such as the absence of signal sequences or the presence of a very short peptide sequence (**Figure 1)**. It is important to note that the 1,158 teretoxin peptides represent a separate, unlabeled dataset. They were not included in the conventional 80/20 training/testing split and thus were not used in computing accuracy, precision, recall, or F1 scores. Instead, MARC predictions on these teretoxins were independently validated using molecular docking, demonstrating the model's applicability to previously unseen, unlabeled data. This step highlights the robustness and novelty of our approach, as positive agreement between model predictions and docking results confirms MARC practical utility.

View Article Online
DOI: 10.1039/D5DD00498E

Table 1. Summary of organisms and molecular targets used in MARC's training and testing sets (sum margins and total).

		Molecular Target				
		Calcium (Ca^{2+}) Channel	**Sodium (Na^{+}) Channel**	**Potassium (K^{+}) Channel**	**Non-Ion Channel**	**TOTAL**
Organism	Cone Snail	**51**	**50**	**17**	**686**	**804**
	Scorpion	**25**	**343**	**307**	**61**	**736**
	Sea Anemone	**0**	**99**	**69**	**72**	**240**
	Snake	**32**	**8**	**41**	**1208**	**1289**
	Spider	**112**	**196**	**110**	**520**	**938**
	TOTAL	**220**	**696**	**544**	**2547**	

Random Forest with ESM features scored highest in determining ion channel targets of venom compounds

We developed a pipeline for testing, training and evaluating MARC's performance. The training set, which made up the majority of the consolidated dataset (80%), was adjusted for class imbalance using SMOTE[44], while ESM[28] and ProtLearn[51] features were applied to generate vectors independently. The test dataset, which is a minority of the consolidated dataset (20%), is used to select the best Random Forest. We evaluated the performance of different Random Forest models using two features: ESM and ProtLearn. The most contributing Prot-Learn features were isolated

View Article Online
DOI: 10.1039/D5DD00498E

with Shapley[56]. MARC's performance was assessed using various metrics, including accuracy, precision, recall, and F1 score (**Figure 2A**). Random Forest ESM performed best on F1-score, precision and recall (~10% for each). XGBoost achieved slightly higher overall accuracy (1-2%), but performed significantly lower on F1-score, precision, and recall when compared to Random Forest ESM. This result suggests that Random Forest provides significantly better class specific sensitivity, whereas XGBoost offers slightly improved general prediction consistency (**Figure 2A**). We also note that ProtLearn-derived features alone provided informative predictions, achieving reasonable classification performance. However, across all benchmarks ESM-derived features consistently outperformed ProtLearn, which is why our main analysis focuses on ESM results (**Figure 2A**). ProtLearn features nevertheless capture useful biochemical and sequence-order information, underscoring their potential interpretability. Direct benchmarking against external venom prediction models was not possible, as existing tools do not address either the venom compounds and ion channel classification problem defined here or the dataset composition used here. Our comparisons therefore focus on alternative feature representations and classifiers within MARC itself.

For downstream applications such as deorphanizing novel venom compounds for identifying potential K^+ channel targeting ones, minimizing false negatives is critical. In such cases, Random Forest's stronger recall makes it a more suitable choice, as it is less likely to miss a candidate venom compound targeting specific ion channels. Random Forest with ESM feature was used to predict the interaction of 1,158 novel teretoxin venom peptides whose ion channel molecular targets are uncharacterized. MARC predictions for teretoxin molecular targets are as follows: 28 peptides predicted to target K^+ ion channels, 971 peptides predicted to be non-ion channel targeting, and 159 peptides were unclassified and labelled "no prediction" **(Figure 2B)**. The majority of teretoxins were predicted to target non-ion channels, followed by a group that MARC could not classify. This finding suggests terebrid venom compounds may are largely not ion channel targeting. However, MARC identified 28 teretoxins that target K^+ channels with a high probability of accuracy. We utilized Receiver Operating Characteristic (ROC) [63] curve analysis to evaluate the predictive performance of our classification model MARC across the four distinct ion channel classes (Na^+, K^+, Ca^{2+}, Non-Ion Channel) (**Figure 2C**).

View Article Online
DOI: 10.1039/D5DD00498E

The ROC curves shown in Figure 2C were computed using the same held-out test set (20% of 4007) used for accuracy, precision, recall, and F1-score evaluations. To evaluate classification performance across four ion channel classes, we computed both per class and micro average ROC curves as described in the Methods section. To assess the impact of including cone snail peptides in the dataset, we generated two ROC plots, one based on the dataset excluding cone snail sequences and one including them. The initial analysis (without cone snail) yielded a high average AUC score of 0.98 (**Figure 2B right & Supplementary Figure 3**), indicating excellent predictive power. Upon inclusion of cone snail peptides, the average AUC decreased to 0.80 (**Figure 2B left**). This drop may reflect the structural and evolutionary divergence of cone snail venom peptides from those of other taxa. Importantly, this performance discrepancy does not reflect overfitting, which would manifest as very high training accuracy and low-test accuracy — a pattern we did not observe. A likely explanation for the reduced accuracy on cone snail peptides is that they possess distinct structural characteristics relative to other taxa, which increases the challenge for prediction. The ROC plots for individual animal groups (**Supplementary Figure 8**) reveal that MARC performs best for scorpion, snake, and sea anemone venom compounds, while performance is slightly lower for spider and cone snail venom compounds, clarifying the source of the observed drop in overall AUC. Also, class specific performance for Ca^{2+} targeting compounds remains comparatively lower, likely due to limited diversity and small sample size in this class, which can reduce recall even after SMOTE balancing. However, MARC maintains robust performance overall, suggesting it generalizes well across diverse venom peptide and protein structures. To further investigate structural factors influencing model performance, we evaluated models trained on full precursor sequences versus mature active peptides alone. Contrary to standard expectations, training on full sequences yielded slightly superior performance. This outcome is driven by the multi-head self-attention mechanisms of the ESM-2 Transformer architecture, which prioritizes bioactive mature regions while simultaneously extracting subtle, evolutionary context from the highly conserved signal and pro-peptides. However, due to the inconsistent availability of signal sequences in proteomic databases, our dataset necessarily comprises a mix of full precursors and mature-only sequences, underscoring MARC's ability to generalize across varying sequence lengths.

View Article Online
DOI: 10.1039/D5DD00498E

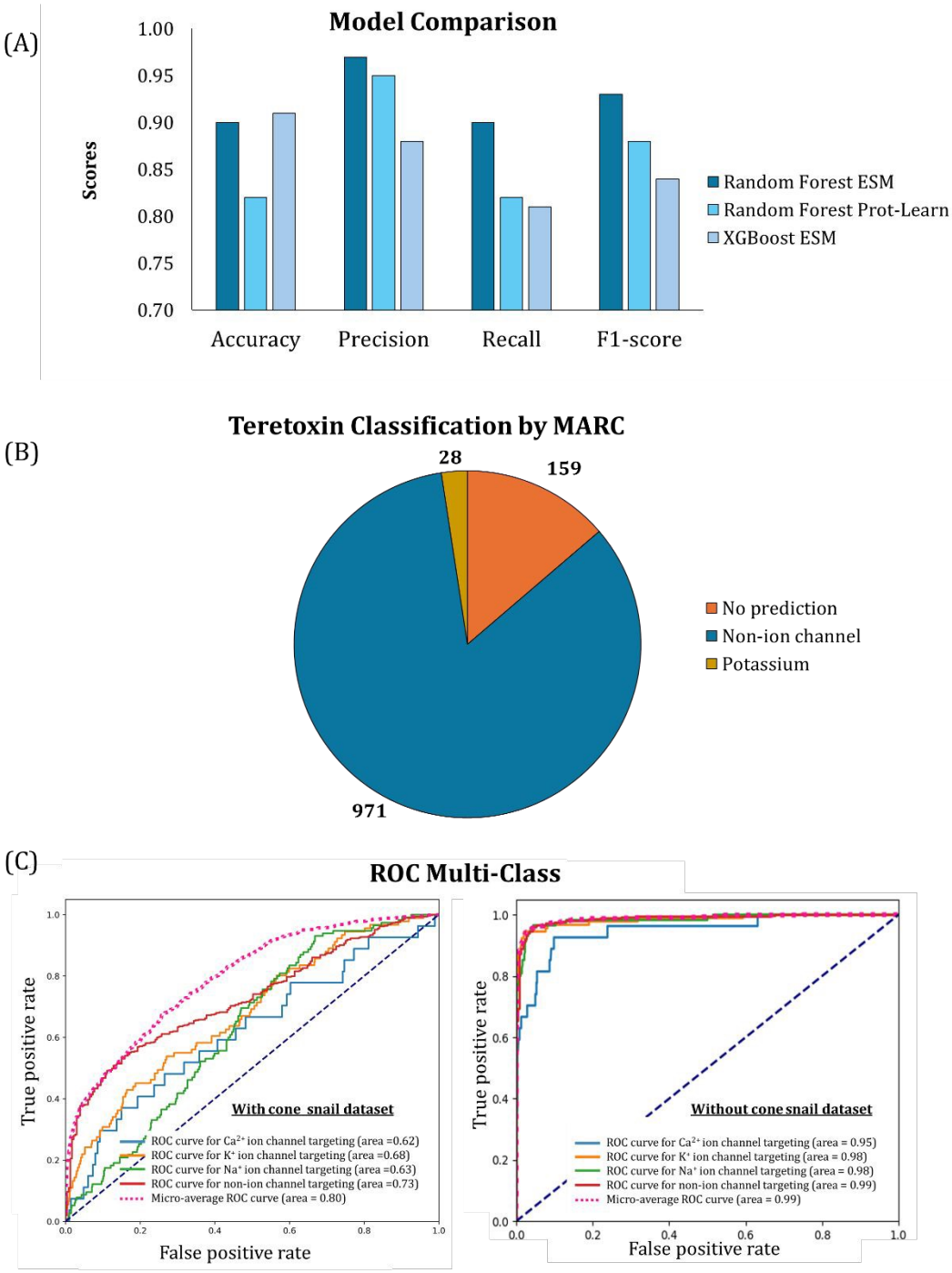


Figure 2. MARC's prediction performance scored best with Random Model using ESM features. (A) Comparison of MARC's accuracy, precision, recall and F1 scores between ESM in the Random Forest versus XGBoost model and the random forest Prot-Learn model. (B) MARC classified a large percentage of teretoxins as non-ion channel targeting, whereas the rest were

View Article Online
DOI: 10.1039/D5DD00498E

classified as Potassium ion channel targeting. The "no prediction" class represents peptides that MARC was unable to classify. (C) Receiver operator characteristics (ROC) curves for each of the three ion-channel targeting classes and non-ion channel targeting class, illustrating the model's ability to distinguish between true positives and false positives. The micro-average AUC value of 0.80 (left – with cone dataset) and 0.99 (right – without cone dataset) across all classes indicates a high degree of predictive accuracy.

MARC predictions suggest that a number of uncharacterized teretoxins target potassium channels

MARC was used to predict the ion channel targets of 1,158 novel teretoxin peptides. Among these, 28 teretoxins were identified as targeting K^+ channels. These teretoxins are uncharacterized and do not have experimentally solved structures. We used AlphaFold2 to predict the structures of the three highest-scoring teretoxins predicted to bind K^+ channels, Cje1.9 from the terebrid *Cinguloterebra jenningsi*, Tpu2.9 from *Terebra straminae* and Tfr4.9 from *Terebra textilis* **Figure 3A(i-iii)**. All of the AlphaFold2.0 generated structures contain three disulfide bonds distributed between a beta-turn joined to a helix, a loop region, and a helix to a helix. The similarity in structure of MARC identified teretoxins suggests that they may have similar molecular function and target similar ion channels. A multiple sequence alignment of these teretoxins revealed similar cysteine frameworks and signal sequences across all members of this class (**Supplementary Figure 2**). "Cysteine framework" is a term used to describe the arrangement of cysteines in the mature sequence of a venom peptide or protein[64]. Framework 9 peptides contain 6 conserved cysteines distributed as C-C-C-C-C-C, with hypervariable residues between the cysteines. MARC predicted K^+ channels targeting teretoxins have cysteine Framework 9, based on known cone snail venom peptides whose molecular target is unknown.[65]

To analyze the structural features of K^+ channel targeting venom peptides, we compared the cysteine framework distribution of all venom compounds in our consolidated dataset with that of the teretoxins targeting K^+ channels (**Figure 3B**). Framework 9 was the most abundant, representing 61% of the K^+ channel targeting teretoxins (17 out of 28) and 48% of the total K^+ channel targeting compounds (263 out of 544). Other major conotoxin frameworks observed

View Article Online
DOI: 10.1039/D5DD00498E

included Framework 33 (29%, 8 out of 28), Framework 22 (3%, 1 out of 28), and a novel framework (7%, 2 out of 28). Clustering the K^+ channel targeting teretoxins based on structural similarity and examining the distribution of cysteine frameworks within these groups revealed two major clusters, represented by Frameworks 9, 33, and a novel framework. Cluster 4 (19 out of 28) emerged as the largest and contained the highest proportion of venom peptides with Framework 9 (18 out of 19), mirroring its abundance in the broader class of K^+ channel targeting venom compounds (**Figure 3C**). These findings support the classification of the 28 identified teretoxins as K^+ channel targeting peptides. It should be noted that Clusters 2, 3, and 5 each contain only 1 peptide each and therefore are considered singletons.

View Article Online
DOI: 10.1039/D5DD00498E

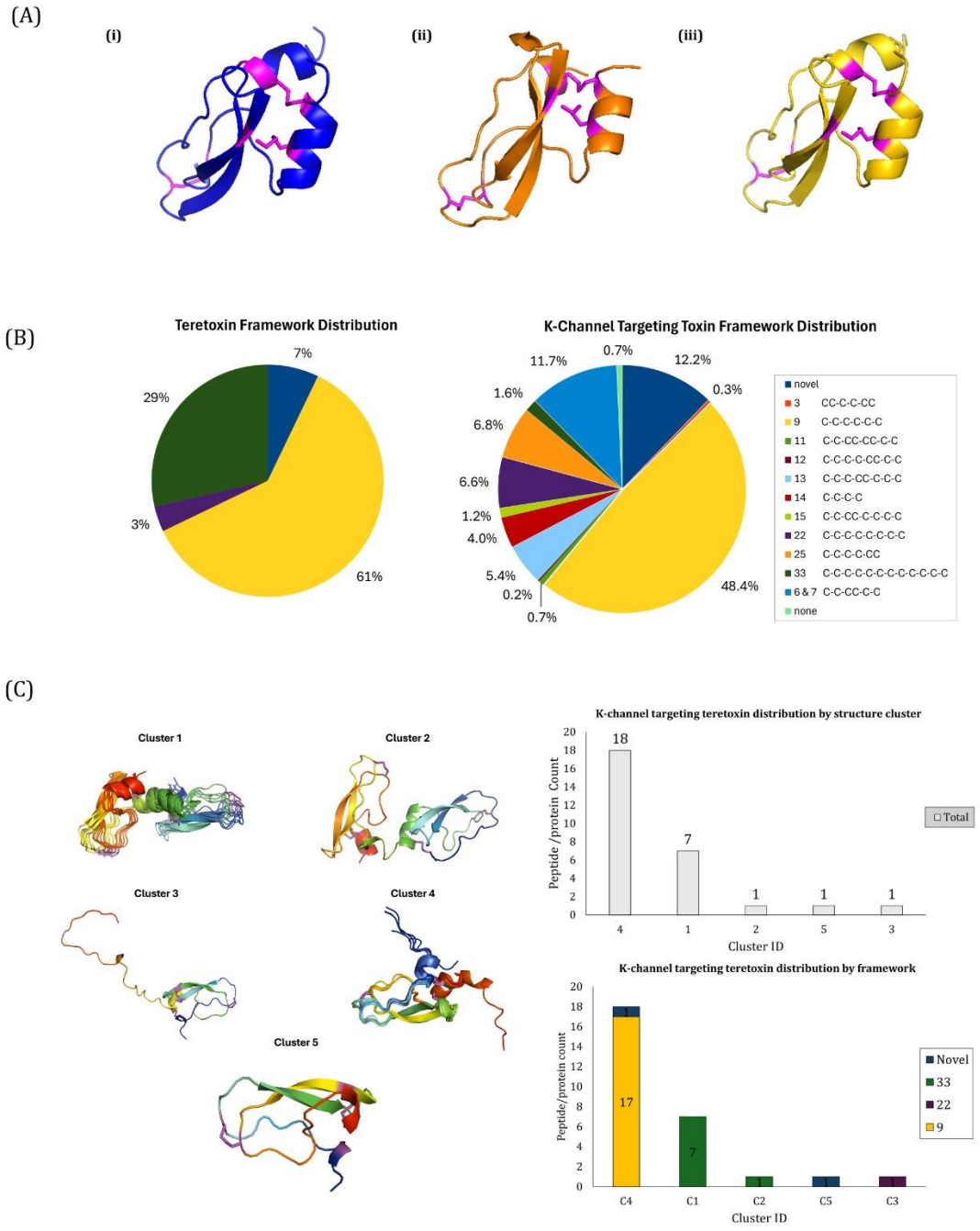


Figure 3. Novel teretoxins predicted to target K^+ channels have similar folds. (A). AlphaFold predictions of the best scoring teretoxins predicted to bind K^+ channels (i) Cje1.9(blue) (ii) Tpu2.9(orange) (iii) Tfr4.9(yellow). Cysteines and disulfide bonds of the teretoxins are highlighted in magenta. (B). Evaluation of the cysteine framework distribution across K^+ channel targeting toxins in the overall dataset (left) and K^+ channel targeting teretoxin dataset (right). The most abundant framework in both groups is framework 9. (C). Distribution of predicted K^+ channel

View Article Online
DOI: 10.1039/D5DD00498E

targeting teretoxins by clusters. Clusters are defined by TM-score using qTMclust from TM-align. A clustering threshold of 0.7 was applied to generate five clusters. Cluster 4 is the largest, consisting predominantly of framework 9 peptides.

View Article Online
DOI: 10.1039/D5DD00498E

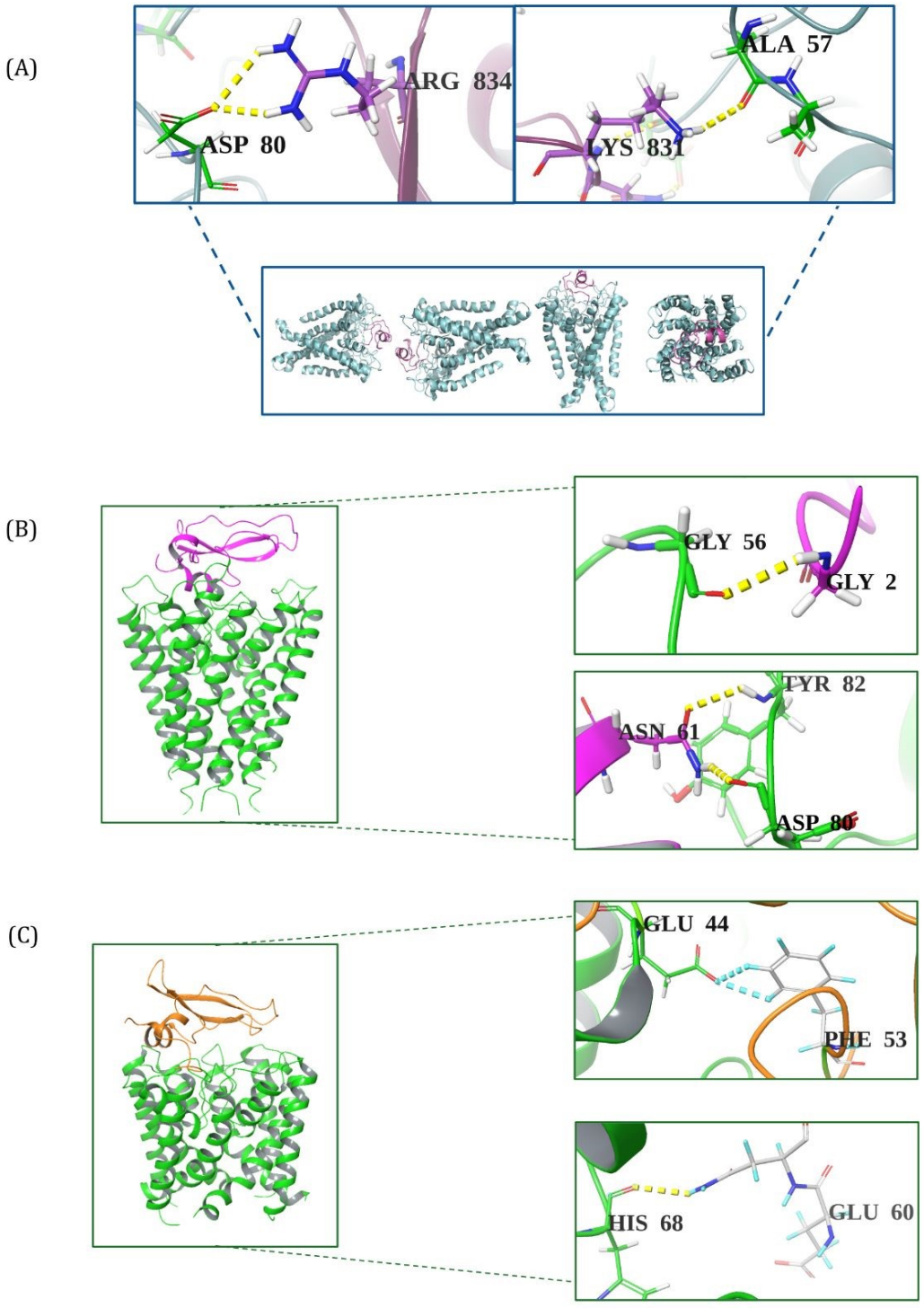


Figure 4. Authentication of MARC-predicted teretoxin:K^+channel interactions using KcSA and MthK potassium channels. (A) Charybdotoxin (magenta, green sticks) in complex with KcSA binding pocket(cyan). Hydrogen bonding interactions between both molecules are also shown. (B) Representative docking pose of Cje1.9 bound to KcsA (Cje1.9:KcsA) (left) and

View Article Online
DOI: 10.1039/D5DD00498E

hydrogen bonding interactions (yellow) observed between KcsA (green) and Cje1.9 (magenta)(left). (C) Representative docking pose (pose_20) of Cje1.9 (orange) bound to 4HYO (green), the MthK potassium channel (Cje1.9:4HYO). Hydrogen bonding (yellow) and aromatic hydrogen bonding (blue) interactions are highlighted (left).

In silico docking and molecular dynamics simulations verify MARC teretoxin: K^+ channel interactions

Orthogonal validation with comparative molecular docking and molecular dynamics simulations was performed to corroborate the robustness of MARC's predictions. We conducted molecular docking and molecular dynamics (MD) simulation experiments to visualize the potential interactions between the highest ranked predicted teretoxin venom peptide, Cje1.9, along with two other teretoxins, Tar2.9 and Hso51_novel (predicted to target K^+ channels), and K^+ channels, KcsA (PDB ID: 2QTO) and MthK (PDB ID: 4HYO). An additional teretoxin, Mkil.1, predicted by MARC to be non-ion channel targeting was also evaluated in the presence of both K^+ channels as the negative control. Both KcsA and MthK have high resolution structures that can be used to perform docking and MD simulation experiments[66,67]. KcsA was chosen for this analysis due to the presence of an experimentally characterized binding interaction with a scorpion venom compound charybdotoxin[67]. A comparative analysis of the binding dynamics of charybdotoxin:KcsA to Cje1.9:KcsA docked pose can provide insight into the feasibility of the prediction. The stability of the peptide:channel interactions were assessed by comparing the hydrogen bonding interactions of a known peptide:channel complex, like charybdotoxin:KcsA, with those of a MARC-predicted peptide:channel complex, Cje1.9:KcsA.

Charybdotoxin:KcsA docking displayed previously reported hydrogen bonding, between the peptide and the ion channel. Specifically, ARG34 (auth: ARG834; charybdotoxin) and ASP80 (KcsA), as well as ALA57 (KcsA) and LYS 31 (auth: LYS 831; charybdotoxin) were identified as key hydrogen bonding partners (**Figure 4A**). While the selected docked pose of Cje1.9:KcsA did not yield hydrogen bonding interactions between similar residues, a few hydrogen bonding interactions between Cje1.9:KcsA were identified in the between GLY56 (KcsA) and GLY2 (Cje1.9) as well as TYR82 (KcsA) and ASN61 (Cje1.9) (**Figure 4B**). The docked pose of Cje1.9:Mthk also yielded hydrogen bonding interactions between HIS68 (MthK) and GLU60

View Article Online
DOI: 10.1039/D5DD00498E

(Cje1.9). Other docked KcsA and MthK complexes also yielded hydrogen bonding interactions which was a requirement for further MD simulations.

A 100 ns MD simulation of the scorpion venom peptide charybdotoxin bound to the K^+ channel KcsA (PDB ID: 2A9H) using the NPγT ensemble represented the known peptide:channel complex which served as the positive control to quantify the hydrogen bonding interactions and Root Mean Square Deviation (RMSD) that indicate a robust interaction. The charybdotoxin:KcsA peptide:channel complex is visualized in both a solvated POPC membrane environment and an unsolvated system. (**Supplementary Figure 6A**). The RMSD for the C-alpha carbons, protein backbone, and the venom peptide (ligand) relative to itself and the protein remained stable, suggesting minimal structural fluctuations and a stable peptide:channel interaction during the simulation (**Supplementary Figure 5A,5B**).Our results revealed stability in the protein backbone and ligand RMSD, suggesting that both the Cje1.9 teretoxin peptide and KcsA ion channel were adequately equilibrated through the course of the simulation (**Supplementary Figure 4A, 5B**).

Additionally, docking Cje1.9 to the potassium ion channel, MthK (PDB ID: 4HYO) established a physiologically stable binding pose[66]. Docking results yielded a single notable pose out of 30 (pose_20), with the most physiologically plausible placement of Cje1.9 outside the membrane (**Figure 4C**). MD simulation of this pose using the NPT ensemble revealed RMSD values consistent with stability across the protein backbone, protein C-alpha carbon atoms, ligand-ligand fit, and ligand-protein fit (**Supplementary Figure 4B, 4C, 5C**). These results suggest that Cje1.9 equilibrated effectively and achieved thermodynamic stability in this configuration and may interact favorably with MthK.

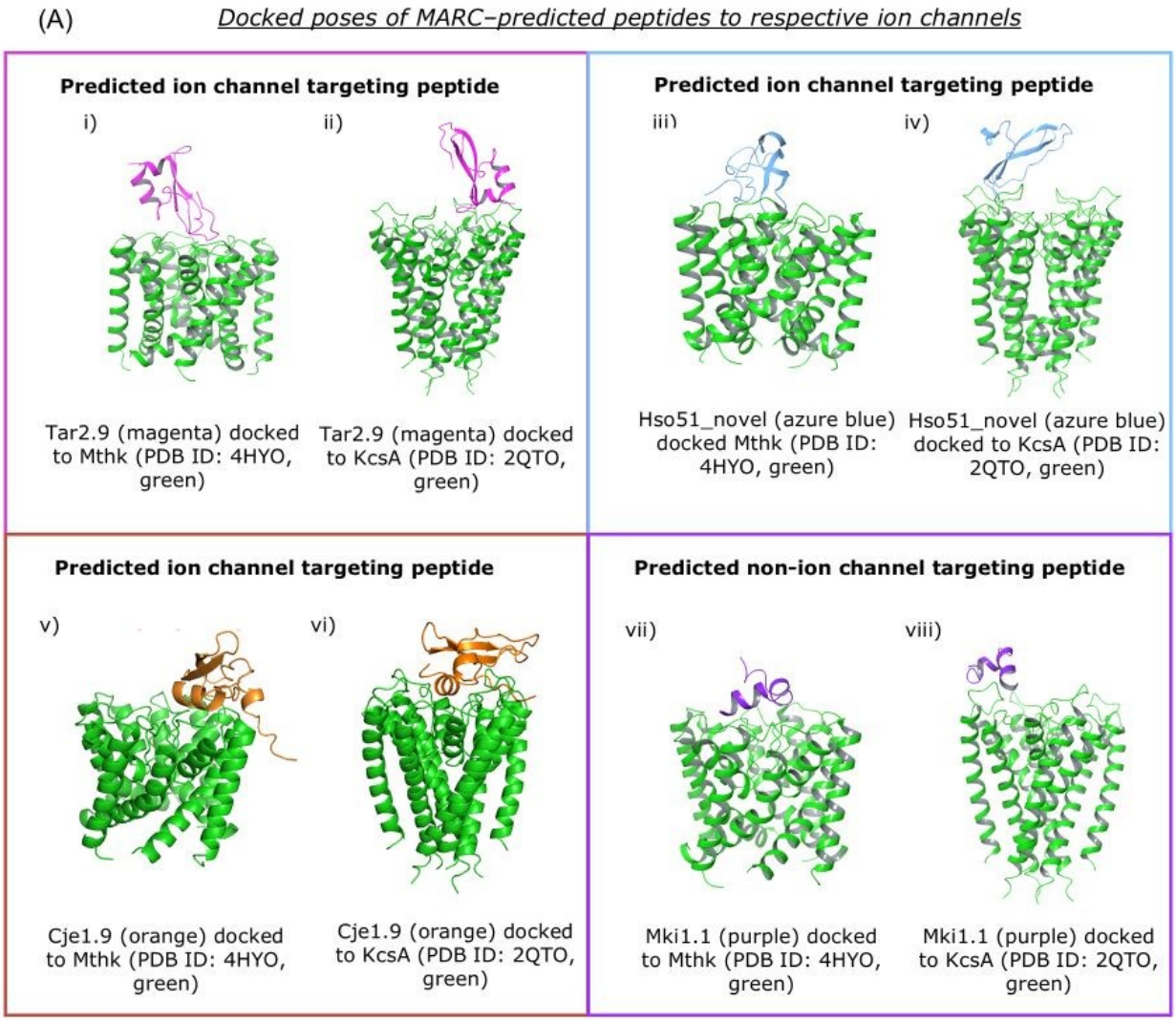
MD simulation parameters for Cje1.9:KcsA, Tar2.9:KcsA, Hso51_novel:KcsA and Mki1.1:KcsA were kept consistent with those used for Charybdotoxin:KcsA (2A9H) to ensure comparability. Cje1.9, Tar2.9 and Hso51_novel, maintained stability and did not diffuse from the binding pocket (**Figure 5A, Supplementary Figure 4A, B, Supplementary Folder 1**) when compared to the negative control Mki1.1:KcsA. The calculated average RMSD for the teretoxin:KcsA complexes evaluated indicated stable interactions with low standard deviation similar to charybdotoxin:KcsA (2A9H, positive control). Specifically, Cje1.9:KcsA, Tar2.9:KcsA, Hso51_nove: KcsA had average RMSD values of 3.0, 4.8, 5.8 respectively compared to the average RMSD of charybdotoxin:KcsA at 6.6 (**Figure 5B**). The negative control, MKi1.1:KcsA demonstrated

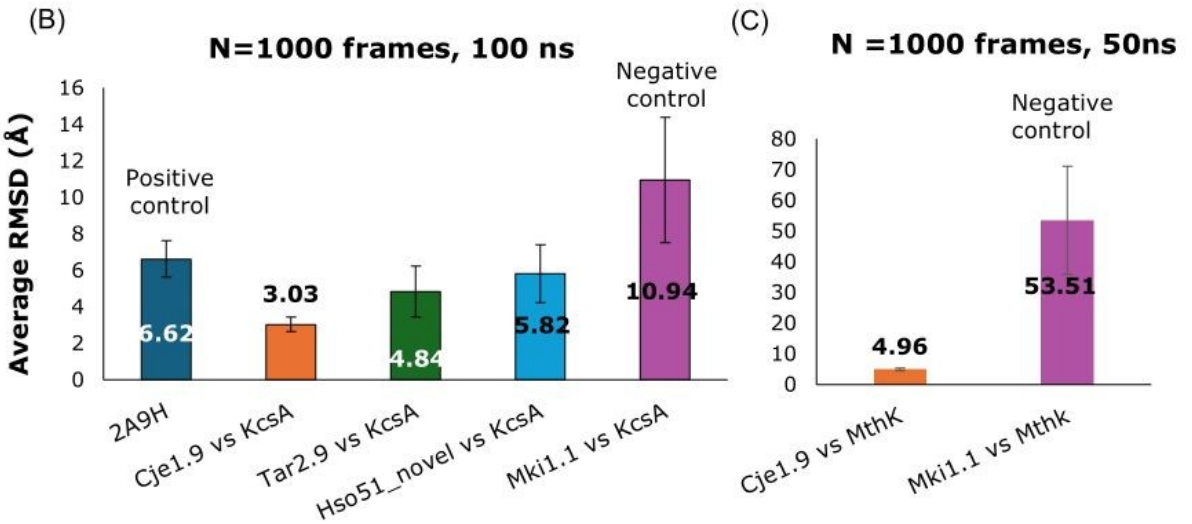
View Article Online
DOI: 10.1039/D5DD00498E

substantial instability with an average RMSD of 10.9 (**Figure 5C**). This finding further confirmed the stability of the teretoxin docked complexes providing support for the MARC predicted K^+ channel modulators (Cje1.9, Tar2.9 and Hso51_novel).

Comparatively, the average RMSD of the docked complex of our highest MARC predicted teretoxin Cje1.9 to MthK[66] (Cje1.9:MthK) was 4.9 compared to the negative control, Mki1.1:Mthk at 53.5, a significant difference between K+ channel predicted vs non-ion channel acting predicted teretoxins (**Figure 5C)**. We further evaluated the hydrogen bonding interaction throughout the simulation for all the complexes evaluated. Our data suggest that all teretoxin:K^+ channel pairs maintain a numerically higher average number of hydrogen bonds than the negative control for both the KcsA and MthK complexes (**Supplementary Figure 6A & B**). While experimental confirmation of the peptide:K-channel interactions are needed, our results indicate a stable interaction between the MARC predicted teretoxins and K-channels. These findings support the accuracy of MARC K^+ ion channel acting teretoxins.

View Article Online
DOI: 10.1039/D5DD00498E





View Article Online
DOI: 10.1039/D5DD00498E

Figure 5. Additional validation of MARC's predictions via docking and molecular dynamics (A) Docked poses of MARC predicted teretoxins peptides to K^+ ion channels: (i) Tar2.9 to MthK (PDB ID: 4HYO) and (ii) KcsA (PDB ID: 2QTO); (iii) Hso51_novel to MthK (4HYO) and (iv) KcsA (2QTO); (v) Cje1.9 to MthK (4HYO) and (vi) KcsA (2QTO); and non-ion channel targeting peptide (iii) Mki1.1 to MthK (4HYO) and (iv) KcsA (2QTO). (B) Average RMSD plot of 100 ns molecular dynamics simulation for all peptide:channel docked poses with charybdotoxin:KcsA (PDB ID: 2A9H) as positive control and Mki1.1 as negative control highlighting stability of interaction between ion-channel targeting peptides and KcsA. (C) 50 ns plot on the right shows the stability of the interaction between Cje1.9, highest MARC predicted K+ Channel teretoxin and Mki1.1 non-ion channel negative control with K+ channel MthK (4HYO).

Conclusion

We developed a new machine learning approach, MARC (Molecular Arms Race Classifier), that mimics the evolutionary adaptions of arms-race predator-prey interactions. MARC is designed to facilitate the characterization of large datasets of venom compounds with unknown ion channel targets. We used MARC to predict the interactions between uncharacterized terebrid snail venom compounds (teretoxins) and specific ion channels (K^+, Na^+ and Ca^{2+}). MARC was trained using a multi-class classifier on sequences of 3205 (out of 4007; 80%) and tested on 801 (out of 4007; 20%) benchmarked venom compounds confirmed to interact with Na^+, K^+, and Ca^{2+} channels or non-ion channel receptors. On an additional dataset of 1158 teretoxin with unknown targets, MARC predicted with high precision that 28 teretoxin venom compounds are most likely to target K^+ Channels. Orthogonal validation of MARC's predictions was achieved using molecular docking and dynamics, which confirmed that the highest predicted novel teretoxin Cje1.9, stably interacts with K^+ channels, KcsA and MthK. This finding demonstrates the utility of MARC in predicting the molecular function of novel venom compounds. This also highlights MARC's application in the identification of potential protein and peptide ligands that could modulate ion channels to further advance investigation of their molecular interactions.

While prior studies have applied imbalance-handling techniques (e.g., SMOTE) or protein language model embeddings in peptide classification, our work is the first to develop and rigorously evaluate a machine learning framework tailored for ion channel–targeting venom

View Article Online
DOI: 10.1039/D5DD00498E

peptides. The novelty lies in four aspects: (i) we curated and harmonized a large, multi-taxon dataset of 5165 venom peptides with ion-channel and non-ion channel annotations which, to our knowledge, has not been systematically analyzed in this context; (ii) we constructed two complementary predictive pipelines—ESM-2 embeddings with Random Forest and ProtLearn-based biochemical descriptors—to balance predictive power with interpretability; (iii) we predicted the ion channel targets of teretoxins with unknown functional activity creating a robust dataset for potential candidates for downstream bioactivity screening and finally (iv) we introduced an orthogonal validation layer in which high-confidence predictions from the aforementioned teretoxin dataset were corroborated through structural modeling, docking, and molecular dynamics simulations.

By utilizing modern representation learning (ESM) and orthogonal validation, this study delivers not only a predictive model, but also a reproducible workflow for prioritizing novel compounds for experimental follow-up. The insights gleaned from MARC have the potential to accelerate the discovery of novel druggable peptides targeting ion channels, with implications for treating ion channel-related disorders.

Author Contributions

Conceptualization, M.H., W.Q, and A.R.; writing—original draft preparation, F.A., A.B, J.R., M.G.; writing—review and editing, F.A., A.B, J.R., M.G., M.H., W.Q, and A.R.; creation of figures and tables: F.A. (Figures: TOC, 1, 2, 3, 4, 5, SF 2, SF 4, additional files for molecular dynamics and docking on Zenodo), A.B.(Figures TOC, 1, 2, SF 3, 8, 9, additional files for MARC on Zenodo), M.G.(Figure TOC, 1, Table 1, ST 2-4), J.R. (SF 1, ST 1); toxin dataset preparation: M.G., J.R; Zenodo dataset preparation: A.B, F.A; Model training and testing (A.B.), model orthogonal validation (F.A), supervision: M.H., W.Q, and A.R. Codes and commands: Model building (A.B..), structure prediction and clustering (F.A.), Frameworks attribution (M.G.), transcriptome assembly (J.R.). All authors have read and agreed to the published version of the manuscript.

Funding: This research and the open accessibility was funded by the National Institutes of Health – Pioneer Award, grant number 5DP1AT012812 to M.H. This project was (partially) supported by

View Article Online
DOI: 10.1039/D5DD00498E

TUFCCC/HC Regional Comprehensive Cancer Health Disparity Partnership, Award and Pre-pilot Award Number U54 CA221704(5) from the National Cancer Institute of National Institutes of Health (NCI/NIH) to MH and FA. Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the NCI/NIH.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: All datasets generated and analyzed during this study, including training and testing data for the MARC classifier, qTMclust clustering outputs, predicted structures, docking results, and molecular dynamics simulation trajectory files, are openly available on Zenodo at: **10.5281/zenodo.17268224 (version 1) & 10.5281/zenodo.19324217 (version 2)**. The archive includes all scripts, processed files, and system preparation directories required to reproduce the analyses described in this manuscript. To ensure broad accessibility and utility for the venom research community, we have developed a standalone inference pipeline. The GitHub repository (https://github.com/holfordlab/Holford_Lab_MARC) includes the pre-trained MARC model weights (marc_model.pkl), a user-friendly prediction script (predict_marc.py), and a detailed README.md file. This README provides straightforward, step-by-step instructions so that any researcher can easily install the environment on their local computer and immediately begin running predictions on their own custom sequence data, without needing to retrain the model. The archive also includes all scripts, processed files, and system preparation directories required to reproduce the analyses described in this manuscript.

Acknowledgments: We acknowledge the Holford Lab members for comments on the manuscript and specifically, Mehakpreet Kaur who assisted with the design of the table of content graphics. We thank Tara Doma, Hagar Abuzaid, Martin Habib, and Eric Li for their participation in the initial data analysis. This work used Bridges-2 at Pittsburgh Supercomputing Center through allocation BIO230212 from the **Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support** (ACCESS) program, which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. Schrodinger Maestro's license was graciously provided through the Sanders Tri-Institutional Therapeutics Discovery Institute. We also thank Schrodinger for additional license support during the revision phase of the manuscript.

View Article Online
DOI: 10.1039/D5DD00498E

Conflicts of Interest: The authors declare no conflict of interest

References

1 M. L. Holding, J. E. Biardi and H. L. Gibbs, *Proceedings of the Royal Society B: Biological Sciences*, 2016, **283**, 20152841.

2 R. Dawkins and J. R. Krebs, *Proc. R. Soc. Lond. B Biol. Sci.*, 1979, **205**, 489–511.

3 V. Schendel, L. D. Rash, R. A. Jenner and E. A. B. Undheim, *Toxins (Basel).*, 2019, **11**, 666.

4 F. Achimba, B. Faezov, B. Cohen, R. Dunbrack and M. Holford, *Mol. Cancer Ther.*, 2024, **23**, 139–147.

5 O. F. Harraz and E. Delpire, *Physiol. Rev.*, 2024, **104**, 23–31.

6 R. Santos, O. Ursu, A. Gaulton, A. P. Bento, R. S. Donadi, C. G. Bologa, A. Karlsson, B. Al-Lazikani, A. Hersey, T. I. Oprea and J. P. Overington, *Nat. Rev. Drug Discov.*, 2017, **16**, 19–34.

7 L. Freuville, C. Matthys, L. Quinton and J.-P. Gillet, *Front. Chem.*, DOI:10.3389/fchem.2024.1465459.

8 Y. Angell, M. Holford and W. H. Moos, *Protein Pept. Lett.*, 2018, **25**, 1044–1050.

9 H. Li, Y. Fang, D. Wang, B. Shi and G. J. Thompson, *Nutr. Diabetes*, 2024, **14**, 86.

10 T. N. Griffith, T. A. Docter and E. A. Lumpkin, *The Journal of Neuroscience*, 2019, **39**, 7086–7101.

11 X. Pan, Z. Li, X. Huang, G. Huang, S. Gao, H. Shen, L. Liu, J. Lei and N. Yan, *Science (1979).*, 2019, **363**, 1309–1313.

12 E. Tosti, R. Boni and A. Gallo, *Mar. Drugs*, 2017, **15**, 295.

13 V. Schendel, L. D. Rash, R. A. Jenner and E. A. B. Undheim, *Toxins (Basel).*, 2019, **11**, 666.

14 M. Holford, M. Daly, G. F. King and R. S. Norton, *Science*, DOI:10.1126/science.aau7761.

15 V. Herzig, *Biomedicines*, 2021, **9**, 413.

16 A.-H. Jin, M. Muttenthaler, S. Dutertre, S. W. A. Himaya, Q. Kaas, D. J. Craik, R. J. Lewis and P. F. Alewood, *Chem. Rev.*, 2019, **119**, 11510–11549.

17 A. Bateman, M. J. Martin, S. Orchard, M. Magrane, R. Agivetova, S. Ahmad, E. Alpi, E. H. Bowler-Barnett, R. Britto, B. Bursteinas, H. Bye-A-Jee, R. Coetzee, A. Cukura, A. Da Silva, P. Denny, T. Dogan, T. G. Ebenezer, J. Fan, L. G. Castro, P. Garmiri, G. Georghiou, L. Gonzales, E. Hatton-Ellis, A. Hussein, A. Ignatchenko, G. Insana, R.

View Article Online
DOI: 10.1039/D5DD00498E

Ishtiaq, P. Jokinen, V. Joshi, D. Jyothi, A. Lock, R. Lopez, A. Luciani, J. Luo, Y. Lussi, A. MacDougall, F. Madeira, M. Mahmoudy, M. Menchi, A. Mishra, K. Moulang, A. Nightingale, C. S. Oliveira, S. Pundir, G. Qi, S. Raj, D. Rice, M. R. Lopez, R. Saidi, J. Sampson, T. Sawford, E. Speretta, E. Turner, N. Tyagi, P. Vasudev, V. Volynkin, K. Warner, X. Watkins, R. Zaru, H. Zellner, A. Bridge, S. Poux, N. Redaschi, L. Aimo, G. Argoud-Puy, A. Auchincloss, K. Axelsen, P. Bansal, D. Baratin, M. C. Blatter, J. Bolleman, E. Boutet, L. Breuza, C. Casals-Casas, E. de Castro, K. C. Echioukh, E. Coudert, B. Cuche, M. Doche, D. Dornevil, A. Estreicher, M. L. Famiglietti, M. Feuermann, E. Gasteiger, S. Gehant, V. Gerritsen, A. Gos, N. Gruaz-Gumowski, U. Hinz, C. Hulo, N. Hyka-Nouspikel, F. Jungo, G. Keller, A. Kerhornou, V. Lara, P. Le Mercier, D. Lieberherr, T. Lombardot, X. Martin, P. Masson, A. Morgat, T. B. Neto, S. Paesano, I. Pedruzzi, S. Pilbout, L. Pourcel, M. Pozzato, M. Pruess, C. Rivoire, C. Sigrist, K. Sonesson, A. Stutz, S. Sundaram, M. Tognolli, L. Verbregue, C. H. Wu, C. N. Arighi, L. Arminski, C. Chen, Y. Chen, J. S. Garavelli, H. Huang, K. Laiho, P. McGarvey, D. A. Natale, K. Ross, C. R. Vinayaka, Q. Wang, Y. Wang, L. S. Yeh and J. Zhang, *Nucleic Acids Res.*, DOI:10.1093/nar/gkaa1100.

18 W. A. Kusuma, A. Fadli, R. Fatriani, F. Sofyantoro, D. S. Yudha, K. Lischer, T. R. Nuringtyas, W. A. Putri, Y. A. Purwestri and R. T. Swasono, *Heliyon*, DOI:10.1016/j.heliyon.2023.e21149.

19 Y. Chu, H. Zhang and L. Zhang, *Toxins (Basel).*, DOI:10.3390/toxins14110811.

20 F. Y. Dao, H. Yang, Z. D. Su, W. Yang, Y. Wu, H. Ding, W. Chen, H. Tang and H. Lin, 2017, preprint, DOI: 10.3390/molecules22071057.

21 D. Koua, A. Ebou and S. Dutertre, *Bioinformatics Advances*, DOI:10.1093/bioadv/vbab011.

22 P. Baldi and S. Brunak, *Bioinformatics: the machine learning approach*, MIT Press, Cambridge, MA, USA, 2nd edn., 2001.

23 N. Sapoval, A. Aghazadeh, M. G. Nute, D. A. Antunes, A. Balaji, R. Baraniuk, C. J. Barberan, R. Dannenfelser, C. Dun, M. Edrisi, R. A. L. Elworth, B. Kille, A. Kyrillidis, L. Nakhleh, C. R. Wolfe, Z. Yan, V. Yao and T. J. Treangen, 2022, preprint, DOI: 10.1038/s41467-022-29268-7.

24 S. Saha and G. P. S. Raghava, *Genomics Proteomics Bioinformatics*, 2006, **4**, 42–47.

25 B. Heil, J. Ludwig, H. Lichtenberg-Fraté and T. Lengauer, *Bioinformatics*, 2006, **22**, 1562–1568.

26 K. Han, M. Wang, L. Zhang, Y. Wang, M. Guo, M. Zhao, Q. Zhao, Y. Zhang, N. Zeng and C. Wang, *Front. Genet.*, 2019, **10**, 399.

27 Y.-W. Zhao, Z.-D. Su, W. Yang, H. Lin, W. Chen and H. Tang, *Int. J. Mol. Sci.*, 2017, **18**, 1838.

View Article Online
DOI: 10.1039/D5DD00498E

28 Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido and A. Rives, *Science (1979).*, DOI:10.1126/science.ade2574.

29 E. Nguyen, M. Poli, M. Faizi, A. W. Thomas, C. B. Sykes, M. Wornow, A. Patel, C. Rabideau, S. Massaroli, Y. Bengio, S. Ermon, S. A. Baccus and C. Ré, in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA, 2023.

30 J. Gorson, G. Ramrattan, A. Verdes, E. M. Wright, Y. Kantor, R. R. Srinivasan, R. Musunuri, D. Packer, G. Albano, W. G. Qiu and M. Holford, *Genome Biol. Evol.*, DOI:10.1093/gbe/evv104.

31 A. Verdes, P. Anand, J. Gorson, S. Jannetti, P. Kelly, A. Leffler, D. Simpson, G. Ramrattan and M. Holford, *Toxins (Basel).*, DOI:10.3390/toxins8040117.

32 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, *Nature*, DOI:10.1038/s41586-021-03819-2.

33 Schrödinger Release 2024-4: Maestro, Schrödinger, LLC, New York, NY, 2024.

34 J. Gorson, G. Ramrattan, A. Verdes, E. M. Wright, Y. Kantor, R. Rajaram Srinivasan, R. Musunuri, D. Packer, G. Albano, W.-G. Qiu and M. Holford, *Genome Biol. Evol.*, 2015, **7**, 1761–78.

35 A. Verdes, D. Simpson and M. Holford, *Genome Biol. Evol.*, 2018, **10**, 249–268.

36 A. M. Bolger, M. Lohse and B. Usadel, *Bioinformatics*, DOI:10.1093/bioinformatics/btu170.

37 S. Andrews, 2010, preprint, https://www.bioinformatics.babraham.ac.uk/projects/fastqc/.

38 B. J. Haas, A. Papanicolaou, M. Yassour, M. Grabherr, P. D. Blood, J. Bowden, M. B. Couger, D. Eccles, B. Li, M. Lieber, M. D. Macmanes, M. Ott, J. Orvis, N. Pochet, F. Strozzi, N. Weeks, R. Westerman, T. William, C. N. Dewey, R. Henschel, R. D. Leduc, N. Friedman and A. Regev, *Nat. Protoc.*, DOI:10.1038/nprot.2013.084.

39 M. G. Grabherr, B. J. Haas, M. Yassour, J. Z. Levin, D. A. Thompson, I. Amit, X. Adiconis, L. Fan, R. Raychowdhury, Q. Zeng, Z. Chen, E. Mauceli, N. Hacohen, A. Gnirke, N. Rhind, F. Di Palma, B. W. Birren, C. Nusbaum, K. Lindblad-Toh, N. Friedman and A. Regev, *Nat Biotechnol. Nat Biotechnol*.

40 M. Manni, M. R. Berkeley, M. Seppey and E. M. Zdobnov, *Curr. Protoc.*, DOI:10.1002/cpz1.323.

View Article Online
DOI: 10.1039/D5DD00498E

41 F. A. Simão, R. M. Waterhouse, P. Ioannidis, E. V. Kriventseva and E. M. Zdobnov, *Bioinformatics*, DOI:10.1093/bioinformatics/btv351.

42 B. Haas and A. Papanicolaou, 2018, preprint, https://github.com/TransDecoder/TransDecoder.

43 S. F. Altschul, W. Gish, W. Miller, E. W. Myers and D. J. Lipman, *J. Mol. Biol.*, DOI:10.1016/S0022-2836(05)80360-2.

44 L. Zhang, C. Zhang, R. Gao, R. Yang and Q. Song, *J. Theor. Biol.*, DOI:10.1016/j.jtbi.2016.04.034.

45 H. Hairani and D. Priyanto, *International Journal of Advanced Computer Science and Applications*, DOI:10.14569/IJACSA.2023.0140864.

46 G. Kim and H. Chun, *BMC Bioinformatics*, 2023, **24**, 432.

47 T. Sainburg, L. McInnes and T. Q. Gentner, *Neural Comput.*, 2021, 1–27.

48 R. Gupta, *Информатика. Экономика. Управление - Informatics. Economics. Management*, 2024, **3**, 0311–0320.

49 A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, D. Bhowmik and B. Rost, *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022, **44**, 7112–7127.

50 A. Elnaggar, H. Essam, W. Salah-Eldin, W. Moustafa, M. Elkerdawy, C. Rochereau and B. Rost, .

51 T. Dorfer, *GitHub*, 2021, preprint, GitHub:0.0.3, https://github.com/tadorfer/protlearn.

52 G. Biau and E. Scornet, *TEST*, 2016, **25**, 197–227.

53 J. Chen, X. Xi and G. Xu, *Smart Cities*, 2025, **8**, 135.

54 S. S. Firoozabadi, M. Ansari and F. Vasheghanifarahani, *European Journal of Business and Management Research*, 2024, **9**, 6–13.

55 T. Chen and C. Guestrin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, 2016, pp. 785–794.

56 B. Rozemberczki, L. Watson, P. Bayer, H.-T. Yang, O. Kiss, S. Nilsson and R. Sarkar, in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence Organization, California, 2022, pp. 5572–5579.

57 S. T. Brown, P. Buitrago, E. Hanna, S. Sanielevici, R. Scibek and N. A. Nystrom, in *Practice and Experience in Advanced Research Computing*, ACM, New York, NY, USA, 2021, pp. 1–4.

View Article Online
DOI: 10.1039/D5DD00498E

58 T. J. Boerner, S. Deems, T. R. Furlani, S. L. Knuth and J. Towns, in *Practice and Experience in Advanced Research Computing*, ACM, New York, NY, USA, 2023, pp. 173–176.

59 G. Madhavi Sastry, M. Adzhigirey, T. Day, R. Annabhimoju and W. Sherman, *J. Comput. Aided. Mol. Des.*, 2013, **27**, 221–234.

60 D. Kozakov, R. Brenke, S. R. Comeau and S. Vajda, *Proteins: Structure, Function, and Bioinformatics*, 2006, **65**, 392–406.

61 C. Lu, C. Wu, D. Ghoreishi, W. Chen, L. Wang, W. Damm, G. A. Ross, M. K. Dahlgren, E. Russell, C. D. Von Bargen, R. Abel, R. A. Friesner and E. D. Harder, *J. Chem. Theory Comput.*, 2021, **17**, 4291–4300.

62 K. J. Bowers, D. E. Chow, H. Xu, R. O. Dror, M. P. Eastwood, B. A. Gregersen, J. L. Klepeis, I. Kolossvary, M. A. Moraes, F. D. Sacerdoti, J. K. Salmon, Y. Shan and D. E. Shaw, in *SC '06: Proceedings of the 2006 ACM/IEEE Conference on Supercomputing*, 2006, p. 43.

63 T. Fawcett, *Pattern Recognit. Lett.*, 2006, **27**, 861–874.

64 V. Lavergne, I. Harliwong, A. Jones, D. Miller, R. J. Taft and P. F. Alewood, *Proceedings of the National Academy of Sciences*, DOI:10.1073/pnas.1501334112.

65 S. D. Robinson and R. S. Norton, *Mar. Drugs*, 2014, **12**, 6058–101.

66 D. J. Posson, J. G. McCoy and C. M. Nimigean, *Nat. Struct. Mol. Biol.*, 2013, **20**, 159–166.

67 L. Yu, C. Sun, D. Song, J. Shen, N. Xu, A. Gunasekera, P. J. Hajduk and E. T. Olejniczak, *Biochemistry*, DOI:10.1021/bi051656d.

View Article Online
DOI: 10.1039/D5DD00498E

Data Availability Statement: All datasets generated and analyzed during this study, including training and testing data for the MARC classifier, qTMclust clustering outputs, predicted structures, docking results, and molecular dynamics simulation trajectory files, are openly available on Zenodo at: **10.5281/zenodo.17268224 (version 1) & 10.5281/zenodo.19324217 (version 2)**. The archive includes all scripts, processed files, and system preparation directories required to reproduce the analyses described in this manuscript. To ensure broad accessibility and utility for the venom research community, we have developed a standalone inference pipeline. The GitHub repository (linked via Zenodo) includes the pre-trained MARC model weights (marc_model.pkl), a user-friendly prediction script (predict_marc.py), and a detailed README.md file. This README provides straightforward, step-by-step instructions so that any researcher can easily install the environment on their local computer and immediately begin running predictions on their own custom sequence data, without needing to retrain the model. The archive also includes all scripts, processed files, and system preparation directories required to reproduce the analyses described in this manuscript.