JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

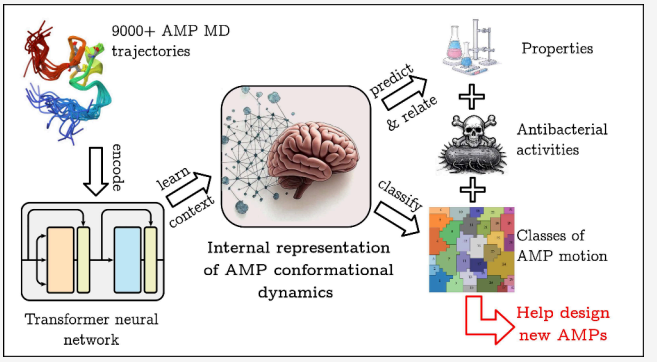
Motion as a Language: Transformer-Based Classification of Antimicrobial Peptide Conformational Dynamics

Benjamin Bouvier*

Cite This: https://doi.org/10.1021/acs.jctc.5c01690 Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: Antimicrobial peptides (AMPs) represent a promising alternative to traditional antibiotics against which many bacteria are rapidly gaining resistance. Today, databases containing tens of thousands of AMPs, along with their properties and biological activities, can be screened to select lead candidates for a given application. The conformational plasticity of AMPs has been proven crucial for the recognition of their targets. However, the volume, complexity, and recalcitrance to classification of conformational data, obtained from e.g. molecular dynamics (MD) simulations, prevent it from being included in databases, let alone used as a criterion for the screening of AMPs. This work applies the transformer neural network architecture (which powers large language models such as ChatGPT) to the detection of temporal and spatial context in time series of AMP conformations from MD simulations. It shows how the representation of AMP conformational space learned by the network can be leveraged for the unsupervised classification of AMP plasticity, which can subsequently be used alongside conventional properties for the screening of databases. Thus, it reveals how deep learning can pave the way toward restoring conformational dynamics to its legitimate importance within drug design pipelines.



INTRODUCTION

Antimicrobial resistance in bacteria poses a growing threat to global public health.[1] On the one hand, the rapid onset of resistance discourages the pharmaceutical industry from investing in the research for new candidates within existing antibiotic classes;[2] on the other, only two truly new classes have reached the market in the last 70 years.[3] In addition, the COVID-19 pandemic has recently provided a painful reminder of how secondary bacterial infections can act as comorbidities to complicate other diseases.[4] The search for alternate approaches to traditional antibiotics is therefore more crucial than ever.

Antimicrobial peptides (AMPs), which have evolved to provide a natural protection against pathogens in organisms spanning every kingdom, represent a promising alternative to conventional antibiotics.[5] AMPs mostly target the bacterial membrane using multiple modes of action, making it difficult for the pathogen to acquire resistance;[6] they often also double as antiviral, anticancer or immunomodulatory agents.[7] Despite the rapidly growing number of identified AMPs − to date, the ADAPTABLE Web server hosted by our research group,[8] which aggregates data from multiple databases, contains more than 40,000 nonredundant AMP sequences − very few AMPs have made it to clinical trials, essentially due to human toxicity and limited absorption issues.[9] However, synthetic AMP derivatives mitigating these issues are now actively researched.[10,11]

In this context, the proficient selection of AMPs as lead compounds for a given application is crucial. Databases collating AMP features and properties are continually enriched and have become invaluable tools for the initial screening process.[8,12−17] However, as for most classes of biomolecules, the definition of a minimal, nonredundant set of properties on which to base an AMP selection and classification process remains an elusive task. While sequence information, chemical properties and biological context (antimicrobial activities, source, modifications···) are the most straightforward descriptors, select databases also include information from clinical trials,[13] genomics and transcriptomics,[12] or structure and conformational dynamics.[16] The latter is of particular interest: indeed, advances in nuclear magnetic resonance and molecular dynamics (MD) simulations have revealed the importance of the molecular plasticity of AMPs in the recognition of their targets.[18] Consequently, MD simulations, which can render both AMP flexibility and membrane lipid dynamics, are now gaining traction as part of antibiotics discovery pipelines.[19] However, because of the difficulty of concisely representing and categorizing the raw conformational data from MD, such simulations are not used in

Received: October 8, 2025
Revised: December 5, 2025
Accepted: January 14, 2026

the AMP screening process, but later in the pipeline as a confirmation of the relevance of previously selected lead AMPs. This is also the case for studies claiming to combine machine learning and MD simulations for the selection of AMPs, which in fact only use MD downstream of the AMP selection process.[2,20]

This paper applies deep learning methods originating from large language models (LLMs) to represent and classify the conformational space of AMPs, as obtained from MD simulations. LLMs based on the transformer architecture[21] and trained on large text corpuses power chatbots (such as ChatGPT) that are currently taking the world by storm. The transformer's success is due to its ability to capture context within texts: if one considers the temporal sequence of AMP conformations and the spatial sequence of amino acids along the AMP chain as a succession of tokens, just like the words in a sentence or paragraph, then the transformer should in theory be able to extract context from AMP conformational data and build an internal representation of it that can be used for many downstream applications, including nonsupervised classification. This work verifies this assumption and shows how such a classification can be used as an additional discriminant for the selection of AMPs, alongside the more traditional features mentioned above, within a drug design pipeline.

METHODS

AMP Conformation Data Set

Aside from being one of the most referenced AMP databases,[22] DBAASP[16] is the state-of-the-art repository of MD trajectories of AMPs simulated under unified conditions (see the previously cited paper for details of the simulation pipeline). The sampling provided by its CHARMM-formatted 400 ns trajectories of solvated AMPs (2000 frames at 0.2 ns intervals) is not exhaustive, but sufficient to provide a reasonable overview of the main attraction basins on the AMP conformational landscapes. 9440 DBAASP entries featured MD data; to unify the length of the time sequences in the data set, only the 9424 entries with full-length trajectories were retained. Similarly, a maximum value for the amino acid sequence lengths of all AMPs in the data set was chosen, with shorter AMPs being padded up to this length and longer AMPs discarded from the data set. Using padding allows the transformer neural network to account for multiple sequence length; however, an excessive amount of padding (i.e., a comparable or higher proportion of padded positions compared to effective positions over the entire data set) will severely hamper learning efficiency and possibly degrade the performance of the trained network. The distribution of AMP lengths was thus analyzed to find the most suitable maximum length. It had minimal and maximal values of 1 and 118, respectively, a mean of 17.00, a median of 16 and a standard deviation of 7.16. The cumulative histogram of data set populations showed a marked plateau around a value of 30 (see Supporting Information figure S1); the maximum length was thus chosen as 30 amino acids, which resulted in 9191 AMPs being retained. Because DBAASP is a manually curated database, no further filtering of AMP entries was performed.

The conformation of an AMP was represented using one Ramachandran number per amino acid. Ramachandran numbers are a concise way of paving (ϕ, ψ) Ramachandran space using a single order parameter comprised between 0 and 1.[23] Each data set entry thus consisted of a real-valued, multivariate series of length 30 along the amino acid sequence dimension and 2000 along the temporal dimension. Each series was segmented into 120 pieces, subseries of respective sequence and temporal length 10 and 50 which were flattened to 1D 500-vectors. Each AMP was thus represented by a 500×120 tensor aggregating all 120 subseries vectors (see Supporting Information figure S2 for a schematic representation). Due to the fact that (i) amino acid sequence positions that have been padded to achieve the common length of 30 amino acids contain no meaningful conformational information, and that (ii) past elements in the time sequence cannot be allowed to attend to future elements due to the unidirectional nature of time,[24] certain pairs of tensor elements must not be allowed to participate in the transformer self-attention mechanism. This was implemented by associating boolean mask tensors to the tensors containing the conformational data.

The amino acid sequence of each AMP was encoded using ProtVec.[25] In this approach, a sequence is decomposed into a set of overlapping, 3-amino acid "words" or 3-grams. These are encoded as vectors, maximizing the probability of word sequences observed throughout the Swiss-Prot database. In this work, an encoding dimensionality of 120 was chosen for the 3-grams, matching the dimensionality of the conformational data (and of similar magnitude to the 100-dimensional 3-grams used in the original ProtVec study). Protein language models (pLMs) provide theoretically superior contextual sequence embeddings using transformers.[26] However, performance gains compared to ProtVec would only become apparent in much longer protein chains where long-distance context is paramount; in addition, pLMs require fine-tuning to achieve their true potential,[27] which is computationally costly and would further complexify the existing transformer model, with no guarantee of performance returns for AMPs shorter than 30 amino acids.

AMP Properties Data Set

All physicochemical properties and antimicrobial activities were taken from the DBAASP database. The 12 physicochemical properties (amphiphilicity index, angle subtended by the hydrophobic residues, disordered conformation propensity, isoelectric point, linear moment, net charge, normalized hydrophobic moment, normalized hydrophobicity, penetration depth, propensity to PPII coil, propensity to *in vitro* aggregation, tilt angle) are defined in the founding DBAASP articles;[16,28] they are not meant as a minimum set of independent descriptors, but have been chosen for their ease of computing and their use as predictors of antimicrobial activities.[28] They were present for all 9191 entries. Antimicrobial activities have been manually extracted from the literature by DBAASP curators;[16] thus, the available activity measures and target microbes varied a lot between entries. To strike the best balance between a large data set and one which features a consistent number of activities per entry, the availability of activities across DBAASP entries was explored. Activities on *Escherichia coli*, *Staphylococcus aureus* and *Pseudomonas aeruginosa* were the most frequent, respectively present in 66%, 59% and 43% of entries (with the next most frequent microorganism, *Candida albicans*, at 23%). 76% of AMPs had activities for at least one of these 3 bacteria (78% including *C. albicans*) and 35% of AMPs had activities for all three (14% including *C. albicans*). For activity measures, the picture was more straightforward, with 76% of entries featuring minimum inhibitory concentration (MIC) values (the next most frequent measure, the minimum bactericidal concentration, came in at 10%). The set of 9191 AMPs with conformational information was thus filtered to retain entries having a MIC value for at least one of *E. coli*, *S. aureus* and *P. aeruginosa*. 6447 entries were thus retained.

The physicochemical properties were normalized according to their nature and the shape of their distributions across the data set, using either *z*-score, logarithmic or sine/cosine schemes (see Supporting Information table S1 for details). All values were then scaled to the $[-1, 1]$ range using min-max scaling, and encoded into a 14-dimensional vector (10 scalar properties, and 2 angular properties each stored as sine and cosine). The MIC values stored in the DBAASP database, having heterogeneous units, were converted to μmol L^{-1}. Based on a histogram of activity values over the data set, a threshold activity of 2000 μmol L^{-1} was selected; higher concentrations were clamped to this threshold, which was also attributed to AMPs missing an activity for one or more of the three considered microorganisms. The MIC values were log-normalized, subjected to min-max scaling, and appended to the aforementioned feature vectors, whose final dimensionality was thus 17. The distributions of all 17 normalized properties and activities over the data set, as well as the correlations between them, are shown on Supporting Information figures S3−S5.

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

Neural Networks

Both conformational data and sequence encodings (tensors consisting of 500 120-dimensional vectors) were projected into a higher-dimensional space using two distinct, fully connected layers. A dimensionality of 432 provided the best trade-off between computational cost and validation loss in preliminary tests (data not shown) and was proportional to the number of attention heads that maximized training efficiency (see below). Positional encodings, providing the otherwise order-agnostic transformer network with the position of each element along the amino acid and time sequences, were defined using a set of learnable parameters. Conformational data, sequence and position encoding tensors were added and fed to a transformer encoder, alongside the matching boolean attention masks, as minibatches of 128 elements. The transformer encoder consisted of 5 stacked subencoder blocks, each comprising a three-headed self-attention layer (each head tasked with the processing of 144 tensor dimensions − this choice did not affect the total number of trainable parameters but maximized training efficiency) and a 2048-dimensional feedforward subnetwork (using larger hyperparameters did not provide meaningful performance gains). The transformer encoder output was an internal representation of the input embeddings, of identical dimensions to the latter (500×432).

The transformer was trained on the reconstruction of missing spans of the input conformational data. Random spans in the input tensors, with lengths following a geometric distribution of average length 3 and representing a total of 15% of the tensor elements, were zeroed out. The output of the transformer encoder was projected back to the dimension of the input data (500×120) using a fully connected layer. The training loss consisted in the mean-squared error (MSE) between the original values of the input tensor at the zeroed-out positions and the reconstructed output values at these positions.

The property predictor model consisted in a series of fully connected layers. The first was used to project the transformer encoder output back to the dimension of the transformer input tensor (500×120). The tensor elements corresponding to padded sequence positions were subsequently zeroed out. A block of fully connected layers of decreasing sizes, with ReLU activation functions (several layer depths and sizes were tried − see Results and Discussion and Supporting Information figure S7) was then applied and the output tensor flattened. Finaly, a single fully connected layer with a *tanh* activation function was used to generate a 17-dimensional vector in which each element represented one of the retained properties (see above). The training loss was the MSE between this prediction and the "ground truth" vector containing the actual normalized property values. To limit overfitting, a dropout layer was intercalated before the decreasing-size fully connected layer block during the training process.

The clustering of the input data set and transformer output was performed using self-organizing map (SOM) neural networks,[29] which produce unsupervised 2D representations of a data set while preserving the topological structure of the data. The mapping space consisted of a grid of neurons, each with a weight vector representing the position of the neuron in the data set vector space. During training, the weights of the neurons in the vicinity of each data set vector were adjusted toward the latter according to a training schedule. A total of 625 neurons, arranged in a square grid of size 25×25, were used: this is in line with the usual guidelines for this choice based on the data set size and dimensionality[30] and confirmed by comparing quantization and topographic errors on a series of grid sizes (data not shown). The neighborhood parameter σ was set to 10, minimizing the values of quantization and topographic errors (see Results and Discussion).

The clustering of the trained SOM neurons was performed as follows. The unified distance matrix (UMAT), which represents the relative Euclidean distance between neuron pairs in training data space, was computed. Neighboring matrix elements with small values tend to correspond to clusters, and are delimited by matrix elements of larger values. To identify the clusters, the connected component approach of Hamel et al.,[31] which connects neurons belonging to a cluster along maximum gradient directions on the UMAT, was employed.

Software Used

All deep neural networks were implemented using PyTorch.[32] The SOMs were built using MiniSom.[33] Multiple sequence alignments were performed using ClustalW 2.1 within Biopython 1.85[34] and represented as sequence logos using Logomaker.[35] With highly divergent sequences, alignments can become unreliable and sometimes reveal artifactual patterns. However, in the present case (i) AMPs are short, (ii) AMPs within a given cluster tend to have similar lengths, and (iii) introducing gaps in the alignment was explicitly forbidden. This drastically reduces the number of possible alignments and the risk of spurious conservation patterns. Low sequence identity will thus result in a logo with many small letters and many residue types per position, which denotes low information content and real sequence variability but is not misleading. The remaining plots and graphics were generated with Matplotlib.[36]

RESULTS AND DISCUSSION

Choice of Neural Network Model

The transformer model processes data using an attention mechanism that lets it weigh the importance of each input element relative to all others (i.e., the context of the input sequence). It consists of two subnetworks: an encoder, tasked with converting the input sequence into an internal contextual representation, and a decoder, which generates the output sequence (text generation, translation, etc.). Unlike its predecessor, the recurrent neural network model, which processes tokens sequentially, the transformer is agnostic to the ordering of elements in the sequence: it processes input tokens in parallel and allows for direct connection between any two elements. This self-attention mechanism thus makes it more computationally efficient, better at capturing long-range dependencies, and not as affected by vanishing gradient issues.[21] Initially developed for natural language processing,[37] the transformer has been successfully applied to time series forecasting and classification[24] based on the analogy between elements in a time series and words in a text. However, time series exhibit unique characteristics: the coexistence of short and long-term dependencies, multivariate complexity wherein variables influence one another, noise and nonstationarity that obfuscate underlying patterns, high dimensionality for long sequences, irregularly spaced data points...[38] Although transformer variants have been designed to address these challenges,[39−42] many researchers have shifted the effort from modifying the transformer architecture to how data is presented to the transformer[43−45] − an approach also adopted here.

Multidimensional Context

The AMP conformational data set (detailed in the Methods section) consists of multivariate time series (one variable per amino acid) with context along both time and amino acid sequence; it can be processed either with channel-independent or channel-mixing approaches. In the former, the individual monovariate time series are treated independently. This approach explicitly disregards any interactions between series but has proven effective in certain scenarios, mainly due to a low propensity to overfitting, data volume efficiency, and the adaptability provided by distinct per-channel attention maps.[44] Channel-mixing models, which integrate dependencies among multiple monovariate series, are potentially able to capture cross-dimensional interactions. They are conceptually more complex and computationally less efficient, and while well-designed models often dominate channel-independent models from a performance point of view,[40,43] they can fail spectacularly: cases in which a simplistic, channel-independent linear network was able to outperform a channel-mixing

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

transformer on the prediction of multivariate time series have been documented.[46]

This paper draws inspiration from the channel-independent PatchTST approach[44] which decomposes each time series into patches that are fed independently into the transformer. Here, however, the patches encompass both the sequence and time dimensions, providing the possibility to detect interdimensional correlations. A fully learnable position embedding scheme was used to provide to the transformer the relative position of each feature vector element along both time and sequence dimensions. This strategy has been proven superior to the sinusoidal encodings traditionally used in transformers;[45] besides, it limits computational and memory usage while providing a straightforward approach to channel-mixing and the potential capture of cross-dimensional context. Similarly to Zerveas et al.,[45] the input data is linearly projected onto a higher-dimensional vector space before being fed to the transformer, allowing for greater flexibility in the detection of correlations.

Transformer Training

Tuning pretrained transformers for new tasks is generally much more efficient than training a new model.[47] This work takes this logic one step further, by building an internal representation of the conformational dynamics of AMPs which can then be coupled to distinct downstream subnetworks to perform a wide variety of tasks. For this, only the encoder part of the transformer architecture was used: indeed, the transformer decoder subnetwork was conceived with generative applications in mind (in the case of time series, forecasting) but is unsuited to unsupervised tasks such as classification or regression. The reuse of a pretrained encoder network, pioneered in large language models such as BERT,[37] has been validated for time series by Zerveas et al.[45]

The transformer network was trained to autoregressively regenerate the missing data from input sequences in which spans of consecutive data have been set to zero (see Methods). For benchmarking, a fully connected network, operating on the same conformational data augmented with positional, sequence and time embeddings as the transformer, was trained for a similar purpose. The red plots on Figure 1 show the MSE loss for the reconstructed data as a function of the number of training epochs. With a 41% lower loss, the superiority of the transformer is clearly apparent. Whereas the transformer architecture has been shown not always to be effective for time series forecasting, sometimes performing significantly worse than a simple fully connected network,[46] in the present case its benefit is plain to see. This could be due to the relative importance of context in time series (which is difficult to assess *a priori*): series with strong contextual dependencies would benefit much more from the self-attention mechanism than more stochastic series; the time-resolved conformational dynamics of AMPs would thus fall into the former category rather than the latter. However, the training process is more costly for the transformer, requiring at least 100 epochs before beating the fully connected network and around 300 more epochs for converged training to be achieved.

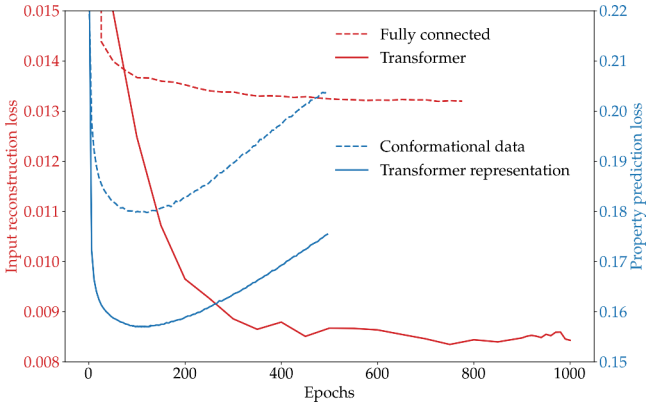


Figure 1. Red plots/left scale: performance of the transformer and fully connected networks for the reconstruction of missing conformational data. Blue plots/right scale: performance of the property predictor network when trained on the conformational data or on the internal representation of the trained transformer.

AMP Property Prediction

I now evaluate the performance gain in using the transformer's internal representation of conformational space as the training set of a predictor network, compared to learning directly on the conformational data itself. To this end, a fully connected predictor network was grafted onto the transformer output and was trained to reproduce the properties and antibacterial activities of the training set AMPs, while keeping the weights of the transformer frozen to their previously trained values. The trained predictor's performance was then compared to that of an identical predictor network trained directly on the conformational data. Since many physicochemical properties from DBAASP have straightforward mathematical definitions and are more convenient to compute directly than to predict using machine learning, this should be seen as a validation of the benefits of the preprocessing of conformational space by the transformer rather than a goal *per se* (this does not apply to antibacterial activities, which can only be measured experimentally and are thus valuable to predict).

As can be seen (blue plots on Figure 1), the trained predictor network achieves a 14% lower loss when applied to the transformer output. This demonstrates a tighter link between the properties of the AMPs and the internal representation compared to the "raw" conformational data, due to the incorporation of temporal context within the former. Looking at the per-property losses (Supporting Information figure S6) reveals 4 to 10-fold prediction performance boosts for properties intuitively related to conformational motion, such as the propensity to *in vitro* aggregation or PPII coils. Interestingly, the performance of the fully connected predictor network was not seen to depend much on the network hyperparameters (number and size of the neuron layers – see Supporting Information figure S7). This suggests a relatively straightforward relationship between the transformer-encoded conformational data and the AMP properties, that even a modestly sized predictor network is able to capture.

Classification of AMPs Based on Conformational Dynamics

As previously discussed, the aim of this work is to provide a reusable, application-agnostic representation of AMP conformational space, in the form of a transformer encoder designed to function as a subnetwork within application-specific modular networks. As an example, I consider the clustering of AMPs based on their conformational dynamics. Associating AMPs with labels summarizing their conformational preferences can usefully complement the typical property and activity data found in AMP databases and provide an additional criterion for the selection of AMPs for a given application. To the best of my knowledge, no database to date provides a classification of peptide motion – DBAASP, the only peptide database to provide conformational dynamics data, does so as "raw"

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

molecular dynamics trajectories which are not directly useable for classification.

The clustering of AMPs based on their conformational dynamics was performed using self-organizing maps (SOMs), applied downstream of the transformer subnetwork. SOMs efficiently project high-dimensional nonlinear data into an intuitive-to-view 2D map, while preserving the data's topological structure (similar high-dimensional vectors stay close together on the map). An unsupervised learning method, it automatically detects patterns in the input data, is not restricted to linear correlations and is robust to noise. To highlight the benefits of the transformer's internal representation, I applied SOMs both to the latter and to the input data set used to train the transformer.

In both cases, a square map of 25 × 25 neurons (a size chosen based on adopted criteria[48]) was used. The neighborhood size σ, defined as the distance from the winning neuron within which neuron weights are adjusted during training, was chosen as the best trade-off between the minimization of quantization error (how faithfully the SOM represents the input data) and topological error (how well the SOM preserves the neighborhood relationships of the input data) − see Supporting Information figure S8. The evolution of both errors with the number of training epochs was also monitored to validate the convergence of the SOM (Supporing Information figure S9). As shown by the topographic error, the trained SOMs were able to adequately capture the neighborhood relationships both for the input data and the transformer inner representation; however, the quantization error was 55% higher for the former, denoting an inability to faithfully transcribe the input data set. This can further be seen on the quality maps, which show how well a neuron in the 2D grid represents the data mapped to it (Figure 2, left panel). The transformer representation exhaustively maps 2D space, with low errors and few neurons left unused. On the other hand, the SOM concentrates all input data into a small proportion of the available neurons, leaving most neurons unused; understandably, the heterogeneity of conformations assigned to a neuron results in high errors. This proves that the transformer was able to disentangle complex trends in the input data, which the downstream SOM then has no difficulty picking; it also shows that these trends are complex enough not to be directly captured by a SOM, despite this model's proven performance on high-dimensional data sets featuring nonlinear correlations.

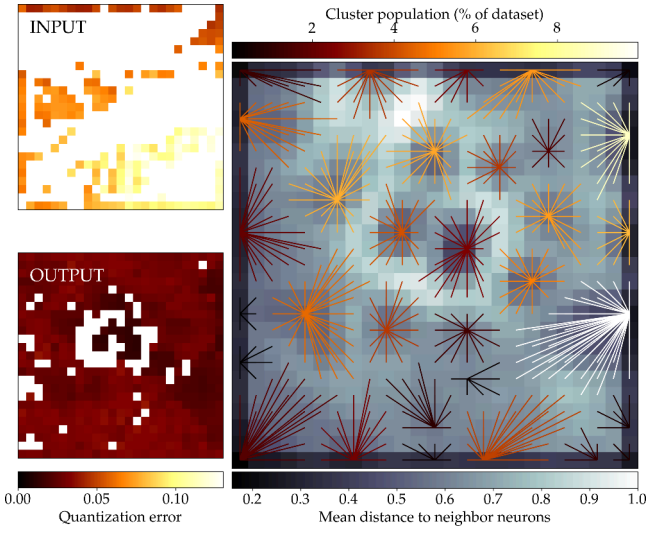


Figure 2. Left: quality maps (mean quantization error per neuron) for the SOM clustering of the input data set (input, top) and the transformer's representation thereof (output, bottom) after training (lower is better, see text for details). Neurons which are not winning neurons for any data set sample are left blank. Right: unified distance matrix (UMAT) for the SOM trained on the transformer internal representation. Contiguous neurons with low distances form clusters; their centers and extents are materialized as starbursts, colored according to the population of the corresponding cluster.

To identify clusters on the SOM array, connected components of SOM neurons were built based on the gradient information in the unified distance matrix (UMAT), as described by Hamel et al.[31] The UMAT, which shows distances between neighboring neurons, is represented on the right panel of Figure 2; on it are superimposed the identified clusters as starbursts. A total of 32 clusters, corresponding to 32 distinct classes of conformational motion in AMPs and with rather homogeneous populations, were found.

To relate conformational dynamics to properties more traditionally used in AMP databases, statistics of physicochemical properties and antimicrobial activities of the AMPs lumped into each SOM cluster were calculated (see Figure 3): a property will be strongly correlated with conformational motion if its distributions show important variations among clusters of conformational dynamics. Unsurprisingly, this is the case for disordered conformation propensity, propensity to PPII coil and propensity to *in vitro* aggregation, which are by their definition directly linked to conformational motion. The amphiphilicity index and the linear moment, both related to the distribution of hydrophobic and hydrophilic amino acids along the peptide chain, also show sizable variations between clusters. Clearly, the accumulation of hydrophobic or hydrophilic residues at the peptide termini (as observed in amphipatic peptides) results in conformational dynamics that strongly differ from that of peptides in which both types of residues are evenly distributed along the chain. Interestingly, the hydrophobic moment is much more homogeneous among clusters, showing that the spatial repartition of residue types on either side of a helix influences conformational dynamics much less than their linear repartition along the amino acid sequence. Tilt angle and penetration depth distributions among clusters are also quite similar, which shows that most AMPs can penetrate membranes regardless of their conformational dynamics − presumably using different mechanisms depending on their amphiphilicity. Finally, for most properties, the transitions between distributions for neighboring clusters are generally rather progressive: this hints at a global correlation in which the similarity between neighboring SOM clusters is carried over to the underlying properties.

For antimicrobial activities, the picture is more complex: activities on *P. aeruginosa* strongly differ between clusters, whereas activities on *S. aureus* are homogeneous; *E. coli* strikes a middle ground. This denotes that an indirect recognition mechanism exists between the *P. aeruginosa* membrane and AMPs, governed by an enhanced selectivity for the conformational dynamics of the latter; conversely, the permeation of the *S. aureus* membrane is conditioned by direct recognition of AMP sequences and/or physicochemical properties. The opposed Gram stains of both bacteria, and the associated differences in membrane architecture, are likely to explain these diverging recognition mechanisms. *E. coli* is Gram-negative like *P. aeruginosa*, but the latter's outer membrane has several unique

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

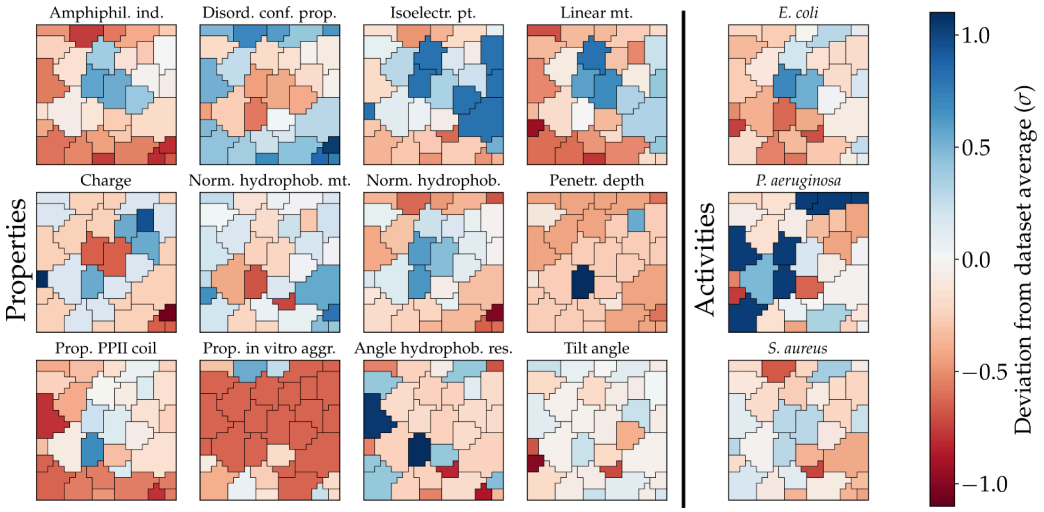


Figure 3. Deviation of the average values of normalized physicochemical properties (left panel) and antimicrobial activities (right panel) of AMPs attributed to each cluster, from the corresponding averages over the entire data set, in units of standard deviation σ. Abbreviations: amphiphil ↔ amphiphilicity, ind ↔ index, isoelectr ↔ isoelectric, pt ↔ point, mt ↔ moment, norm ↔ normalized, hydrophob ↔ hydrophobic/hydrophobicity, penetr ↔ penetration, prop ↔ propensity, aggr ↔ aggregation, res ↔ residues.

features (longer and more highly charged core oligosaccharides,[49] prevalence of penta-acylated lipid A,[50] fewer porins[51]) that make it notoriously more resistant to antibiotics.[52] From the present results, a highly selective membranotropicity toward AMPs featuring certain classes of collective motion can be added to the list of specificities explaining *P. aeruginosa*'s superior resistance capabilities.

Figure 4 shows the correlation (as the absolute value of Pearson's coefficient) between the distributions of the average

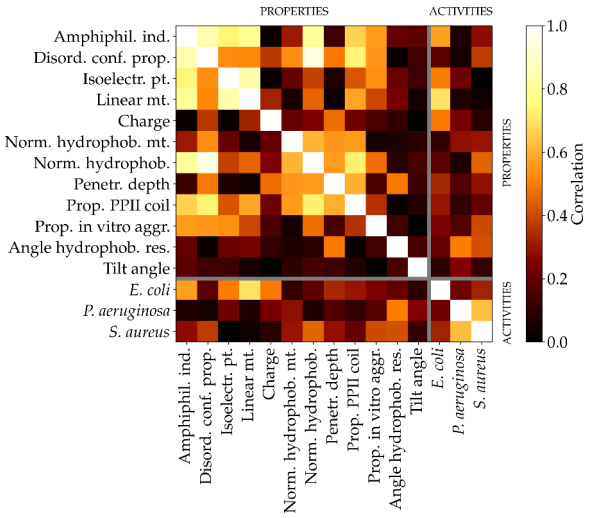


Figure 4. Pairwise similarities between the distributions of the average values of physicochemical properties and antimicrobial activities in SOM neurons, measured as the absolute value of the Pearson correlation coefficient between the corresponding distributions.

values of physicochemical and antimicrobial properties among SOM neurons − in simple terms, the pairwise similarities between the Figure 3 subplots. Amphiphilicity index, isoelectric point and linear moment all feature high correlation with one another, demonstrating that they all similarly influence AMP conformational dynamics. The screening of AMPs for a given application, especially one where conformational dynamics is suspected to play an important role (such as the targeting of *P. aeruginosa*), can be facilitated by retaining only one of these properties in the selection filter. Similarly, the propensity of disordered conformations is also highly correlated with normalized hydrophobicity and amphiphilicity index as far as conformational dynamics are concerned. The rest of the properties show no significant correlation with one another and are thus best kept as independent descriptors. Similarly, the per-cluster distributions of antibacterial activities do not feature marked correlations with any of the distributions of physicochemical properties, confirming that the relationship between conformational motion and activity exists but is quite complex. Interestingly, although the distribution of activities against *S. aureus* in the SOM clusters is rather homogeneous, it displays some correlation with the much more marked distribution of activities against *P. aeruginosa*, denoting a limited form of common indirect recognition of AMPs from the two species despite the differences in their membrane constitutions − AMPs active against one of these bacteria could potentially hold promise against the other.

Finally, I relate the 32 classes of conformational dynamics to the amino acid sequences of the corresponding AMPs. The sequence logos for a selection of representative clusters are shown on Figure 5 (the logos for all clusters are provided on Supporting Information figure S10) − please see the Methods section for a critical assessment of the alignment of highly divergent sequences in some clusters. The first observation is that AMP sequence length is a determinant factor of conformational motion: the median length varies a lot from cluster to cluster and the standard deviation of sequence lengths within a cluster is generally quite small (see Supporting Information figure S10). Clusters of comparable AMP lengths also tend to occupy neighboring positions on the SOM map, which shows that sequence length plays an important role in the global topology of the AMP conformational dynamics hyperspace. Second, many clusters are characterized by conserved sequence motifs: this is especially clear for clusters 12 and 27 (whose respective populations of 384 and 99 are sufficiently large to ensure statistical significance). Proline-rich AMPs have unique dynamic properties and are thus segregated in their own

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

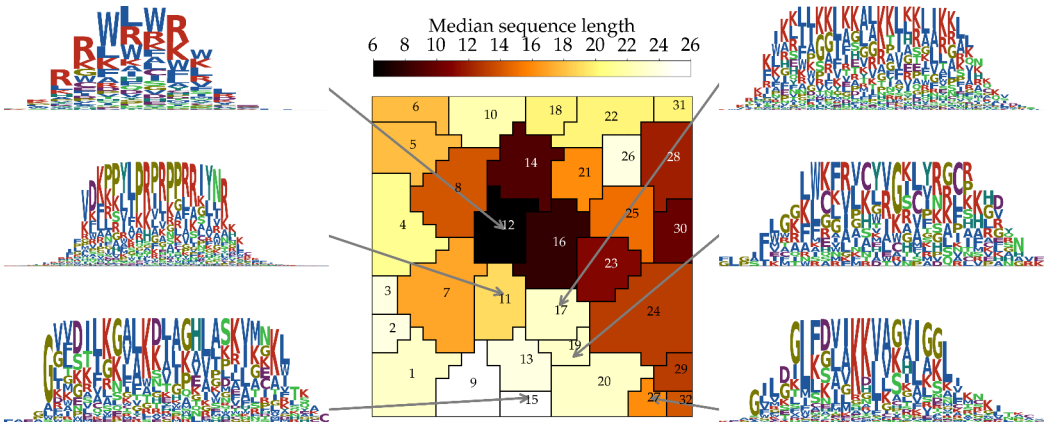


Figure 5. Amino acid sequence logos of AMPs belonging to a selection of clusters on the SOM map (denoted by arrows). The logos use the color scheme of Najafabadi et al.[53] The SOM clusters are colored according to the median sequence length of their associated AMPs. The integer indices identifying the clusters on the map are also used to refer to them in the article text.

cluster 11; similarly, cysteine-containing peptides are concentrated in cluster 19. Spans of repeating KKL motifs also appear to induce specific conformational motion; they appear prominently in several clusters of different median AMP lengths (e.g., cluster 17 for AMPs of length 21−25). Glycine also features prominently at specific positions along the sequence in some clusters (15 for long AMPs, 27 for midsized ones), in particular in the terminal regions; in cluster 27, glycines appear to modulate the effect of KKL motifs. Similarly, arginine-rich peptides apparently have their own conformational behavior; interestingly, in the corresponding clusters (e.g., cluster 12), hydrophobic tryptophan residues and arginines appear interchangeable (in terms of probability of occurrence) at several positions without seemingly affecting conformational motion. Finally, negatively charged amino acids (mostly aspartic acid), though less frequent, are encountered with good probability at certain positions in select clusters (e.g., cluster 15), in which polar uncharged residues (serine, asparagine) can also be found.

Combining Existing AMP Databases and Conformational Dynamics Classes

Using conformational dynamics as an additional criterion to filter AMP databases carries a strong incentive for the design of novel AMPs against specific microbial targets. As an example, I provide in Supporting Information a detailed account of the combination of the ADAPTABLE AMP database[8] with the conformational dynamics classes from this work for the suggestion of putative lead AMPs against *Shigella* sp, and summarize the main findings below.

First, the distribution of the majority of AMPs active against this genus in very few SOM classes demonstrates a strong relationship between plasticity and activity, as observed above for *P. aeruginosa*. Second, the nonuniformity of this distribution can be exploited for the efficient pruning of the ADAPTABLE database − either by only considering as lead candidates the AMPs from the most populated SOM classes, or on the contrary by sampling all classes, selecting a small number of representative AMPs for each. Finally, the examination of consensus sequences within the classes provides three possible paths toward the design of candidate AMPs: (i) recommend known AMPs with suitable dynamics that have not yet been tested on the target; (ii) isolate minimal motifs with distinct plasticity from longer AMPs, to be used as templates to design new peptides; (iii) suggest novel AMPs that, although chimeric and untested, have both the desired conformational behavior and high sequence identity to existing AMPs active on other bacterial targets. Naturally, the actual activities of these candidates on *Shigella* would need to be experimentally tested (which is planned shortly in our lab for cancer-targeting AMPs); until they are, the general guidelines inferred from this example carry more weight than the exact AMP sequences involved. Nonetheless, the trends within and between classes of conformational dynamics and the relationship between activity and motion observed in this work leave little doubt as to the usefulness of augmenting existing AMP databases with conformational data.

CONCLUDING REMARKS

The importance of conformational dynamics in biomolecular recognition has been acknowledged for a long time, but its practical use within drug design pipelines has remained anecdotical. Advances in the application of LLM methods to time series could help resolve this paradox by addressing the infamous intractability of raw conformational data. The unsupervised learning of a classification of AMP dynamics, described herein, is a valuable example. Ideally, a definitive classification would benefit from being trained on many more AMPs (less than 40% of peptides in DBAASP have standardized MD simulation trajectories) and longer simulations; this would involve a sizable computational effort but would not constitute a major conceptual roadblock. Other conformational dynamics-related time series could possibly also be incorporated into the data set without negatively impacting the performance-to-computational cost ratio or causing overfitting. Finally, new developments in transformer networks should also be monitored on a regular basis to take advantage of the rapid advances in the field, notably for the better treatment of long-range and multiscale correlations. We will be investigating these three points in view of the future inclusion of conformational criteria into our ADAPTABLE database. In the meantime, the inner representation of conformational space learned by the transformer will be leveraged on other tasks, by coupling it to *ad hoc* downstream networks: for instance, for the prediction of the conformational motion of peptides and the design of collective variables to efficiently explore conformational space within enhanced sampling MD simulations.

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX

■ ASSOCIATED CONTENT

Data Availability Statement

For purposes of reproducibility, data sets of encoded AMP conformations, properties and activities, source code and trained network parameters are available for download at https://sdrive.cnrs.fr/s/sXR5wzbCF8YAsXd (uncompressed data size: 12 GiB).

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.5c01690.

Additional methodological details: normalization scheme for the properties data set and distributions and correlations thereof; distribution of data set AMP lengths; schematic representation of input tensors; validation of network hyperparameters; per-property MSE loss for the trained predictor network. Statistics and logos of amino acid sequences within all classes of AMP conformational dynamics. Example of the combination of an existing AMP database and said classes for the design of novel AMPs (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Benjamin Bouvier − *Enzyme and Cell Engineering, CNRS UMR7025/Université de Picardie Jules Verne, 80039 Amiens, France;* orcid.org/0000-0001-8782-2426; Email: benjamin.bouvier@u-picardie.fr

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.5c01690

Notes

The author declares no competing financial interest.

■ ACKNOWLEDGMENTS

The calculations presented herein were performed using HPC resources from the UPJV MatriCS computing platform.

■ REFERENCES

(1) World Health Organization *Antimicrobial resistance: global report on surveillance*; 2014.
(2) Ruiz Puentes, P.; Henao, M. C.; Cifuentes, J.; Muñoz-Camargo, C.; Reyes, L. H.; Cruz, J. C.; Arbeláez, P. Rational discovery of antimicrobial peptides by means of artificial intelligence. *Membranes* **2022**, *12*, 708.
(3) Coates, A. R.; Halls, G.; Hu, Y. Novel classes of antibiotics or more of the same? *Br. J. Pharmacol.* **2011**, *163*, 184−194.
(4) Vaillancourt, M.; Jorth, P. The unrecognized threat of secondary bacterial infections with COVID-19. *mBio* **2020**, *11*, e01806-20.
(5) Rima, M.; Rima, M.; Fajloun, Z.; Sabatier, J.-M.; Bechinger, B.; Naas, T. Antimicrobial peptides: a potent alternative to antibiotics. *Antibiotics* **2021**, *10*, 1095.
(6) Abdi, M.; Mirkalantari, S.; Amirmozafari, N. Bacterial resistance to antimicrobial peptides. *J. Pept. Sci.* **2019**, *25*, e3210.
(7) Zhang, D.; He, Y.; Ye, Y.; Ma, Y.; Zhang, P.; Zhu, H.; Xu, N.; Liang, S. Little antimicrobial peptides with big therapeutic roles. *Protein Pept. Lett.* **2019**, *26*, 564−578.
(8) Ramos-Martín, F.; Annaval, T.; Buchoux, S.; Sarazin, C.; D'Amelio, N. ADAPTABLE: a comprehensive web platform of antimicrobial peptides tailored to the user's research. *Life Sci. Alliance* **2019**, *2*, e201900512.
(9) Wang, C.; Hong, T.; Cui, P.; Wang, J.; Xia, J. Antimicrobial peptides towards clinical application: Delivery and formulation. *Adv. Drug Delivery Rev.* **2021**, *175*, 113818.
(10) Kumar, P.; Kizhakkedathu, J.; Straus, S. Antimicrobial peptides: diversity, mechanism of action and strategies to improve the activity and biocompatibility in vivo. *Biomolecules* **2018**, *8*, 4.
(11) Tan, P.; Fu, H.; Ma, X. Design, optimization, and nanotechnology of antimicrobial peptides: From exploration to applications. *Nano Today* **2021**, *39*, 101229.
(12) Yao, L.; Guan, J.; Xie, P.; Chung, C.-R.; Zhao, Z.; Dong, D.; Guo, Y.; Zhang, W.; Deng, J.; Pang, Y.; Liu, Y.; Peng, Y.; Horng, J.-T.; Chiang, Y.-C.; Lee, T.-Y. dbAMP 3.0: updated resource of antimicrobial activity and structural annotation of peptides in the post-pandemic era. *Nucleic Acids Res.* **2025**, *53*, D364−D376.
(13) Ma, T.; Liu, Y.; Yu, B.; Sun, X.; Yao, H.; Hao, C.; Li, J.; Nawaz, M.; Jiang, X.; Lao, X.; Zheng, H. DRAMP 4.0: an open-access data repository dedicated to the clinical translation of antimicrobial peptides. *Nucleic Acids Res.* **2025**, *53*, D403−D410.
(14) Gawde, U.; Chakraborty, S.; Waghu, F. H.; Barai, R. S.; Khanderkar, A.; Indraguru, R.; Shirsat, T.; Idicula-Thomas, S. CAMPR4: a database of natural and synthetic antimicrobial peptides. *Nucleic Acids Res.* **2023**, *51*, D377−D383.
(15) Wang, G.; Zietz, C. M.; Mudgapalli, A.; Wang, S.; Wang, Z. The evolution of the antimicrobial peptide database over 18 years: milestones and new features. *Protein Sci.* **2022**, *31*, 92−106.
(16) Pirtskhalava, M.; Amstrong, A. A.; Grigolava, M.; Chubinidze, M.; Alimbarashvili, E.; Vishnepolsky, B.; Gabrielian, A.; Rosenthal, A.; Hurt, D. E.; Tartakovsky, M. DBAASP v3: database of antimicrobial/cytotoxic activity and structure of peptides as a resource for development of new therapeutics. *Nucleic Acids Res.* **2021**, *49*, D288−D297.
(17) Ye, G.; Wu, H.; Huang, J.; Wang, W.; Ge, K.; Li, G.; Zhong, J.; Huang, Q. LAMP2: a major update of the database linking antimicrobial peptides. *Database* **2020**, *2020*, baaa061.
(18) de Paula, V. S.; Valente, A. P. A dynamic overview of antimicrobial peptides and their complexes. *Molecules* **2018**, *23*, 2040.
(19) Palmer, N.; Maasch, J. R. M. A.; Torres, M. D. T.; de la Fuente-Nunez, C. Molecular dynamics for antimicrobial peptide discovery. *Infect. Immun.* **2021**, *89*, e00703-20.
(20) Cao, Q.; et al. Designing antimicrobial peptides using deep learning and molecular dynamic simulations. *Briefings Bioinf.* **2023**, *24*, bbad058.
(21) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *arXiv* 2017 1706.03762.
(22) Zhang, K.; Teng, D.; Mao, R.; Yang, N.; Hao, Y.; Wang, J. Thinking on the construction of antimicrobial peptide databases: powerful tools for the molecular design and screening. *Int. J. Mol. Sci.* **2023**, *24*, 3134.
(23) Mannige, R. V.; Kundu, J.; Whitelam, S. The Ramachandran number: an order parameter for protein geometry. *PLoS One* **2016**, *11*, e0160023.
(24) Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; Sun, L. Transformers in time series: a survey. *arXiv* 2022 2202.07125.
(25) Asgari, E.; Mofrad, M. R. K. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS One* **2015**, *10*, e0141287.
(26) Bepler, T.; Berger, B. Learning the protein language: evolution, structure, and function. *Cell Syst.* **2021**, *12*, 654−669.
(27) Schmirler, R.; Heinzinger, M.; Rost, B. Fine-tuning protein language models boosts predictions across diverse tasks. *Nat. Commun.* **2024**, *15*, 7407.
(28) Vishnepolsky, B.; Pirtskhalava, M. Prediction of linear cationic antimicrobial peptides based on characteristics responsible for their interaction with the membranes. *J. Chem. Inf. Model.* **2014**, *54*, 1512−1523.
(29) Kohonen, T. Essentials of the self-organizing map. *Neural Netw.* **2013**, *37*, 52−65.

(30) Vesanto, J.; Alhoniemi, E. Clustering of the self-organizing map. *IEEE Trans. Neural Netw.* **2000**, *11*, 586−600.
(31) Hamel, L.; Brown, C. W. Improved interpretability of the unified distance matrix with connected components. DMIN 2011. *proceedings of the 2011 international conference on data mining* **2011**, 338−343.
(32) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; Vito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., D'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc., 2019; pp 8024−8035.
(33) Vettigli, G. MiniSom: minimalistic and NumPy-based implementation of the Self Organizing Map. 2018; https://github.com/JustGlowing/minisom.
(34) Chapman, B.; Chang, J. Biopython: Python tools for computational biology. *ACM SIGBIO Newsl.* **2000**, *20*, 15−19.
(35) Tareen, A.; Kinney, J. B. Logomaker: beautiful sequence logos in Python. *Bioinformatics* **2020**, *36*, 2272−2274.
(36) Hunter, J. D. Matplotlib: a 2D graphics environment. *Comp. Sci. Eng.* **2007**, *9*, 90−95.
(37) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv* 2018, 1810.04805.
(38) Kottapalli, S. R. K.; Hubli, K.; Chandrashekhara, S.; Jain, G.; Hubli, S.; Botla, G.; Doddaiah, R. Foundation models for time series: a survey. *arXiv*. 2025, 2504.04011.
(39) Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: beyond efficient transformer for long sequence time-series forecasting. *arXiv* 2020, 2012.07436.
(40) Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; Long, M. iTransformer: inverted transformers are effective for time series forecasting. *arXiv*. 2023, 2310.06625.
(41) Zhang, Y.; Ma, L.; Pal, S.; Zhang, Y.; Coates, M. Multi-resolution time-series transformer for long-term forecasting. *arXiv*. 2023, 2311.04147.
(42) Chen, P.; Zhang, Y.; Cheng, Y.; Shu, Y.; Wang, Y.; Wen, Q.; Yang, B.; Guo, C. Pathformer: multi-scale transformers with adaptive pathways for time series forecasting. *arXiv*. 2024, 2402.05956.
(43) Wang, H.; Mo, Y.; Xiang, K.; Yin, N.; Dai, H.; Li, B.; Fan, S.; Mo, S. CSformer: combining channel independence and mixing for robust multivariate time series forecasting. *arXiv*. 2023, 2312.06220.
(44) Nie, Y.; Nguyen, N. H.; Sinthong, P.; Kalagnanam, J. A time series is worth 64 words: long-term forecasting with transformers. *arXiv*. 2022, 2211.14730.
(45) Zerveas, G.; Jayaraman, S.; Patel, D.; Bhamidipaty, A.; Eickhoff, C. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2021.
(46) Zeng, A.; Chen, M.; Zhang, L.; Xu, Q. Are transformers effective for time series forecasting? *arXiv*. 2022, 2205.13504.
(47) Liu, J.; Song, Y.; Xue, K.; Sun, H.; Wang, C.; Chen, L.; Jiang, H.; Liang, J.; Ruan, T. FL-tuning: layer tuning for feed-forward network in transformer. *arXiv*. 2022, 2206.15312.
(48) Shalaginov, A.; Franke, K. A new method for an optimal SOM size determination in neuro-fuzzy for the digital forensics applications. *Adv. Comput. Intell.* **2015**, *9095*, 549−563.
(49) Kocincova, D.; Lam, J. S. Structural diversity of the core oligosaccharide domain of Pseudomonas aeruginosa lipopolysaccharide. *Biochem. (Mosc.)* **2011**, *76*, 755−760.
(50) Lam, J. S.; Taylor, V. L.; Islam, S. T.; Hao, Y.; Kocíncová, D. Genetic and functional diversity of Pseudomonas aeruginosa lipopolysaccharide. *Front. Microbiol.* **2011**, *2*, 118.
(51) Zgurskaya, H. I.; López, C. A.; Gnanakaran, S. Permeability barrier of Gram-negative cell envelopes and approaches to bypass it. *ACS Infect. Dis.* **2015**, *1*, 512−522.
(52) Laborda, P.; Hernando-Amado, S.; Martínez, J. L.; Sanz-García, F. Antibiotic Resistance in Pseudomonas. In *Pseudomonas aeruginosa: Biology, Pathogenesis and Control Strategies*; Springer International Publishing, 2022; p 117−143.
(53) Najafabadi, H. S.; Garton, M.; Weirauch, M. T.; Mnaimneh, S.; Yang, A.; Kim, P. M.; Hughes, T. R. Non-base-contacting residues enable kaleidoscopic evolution of metazoan C2H2 zinc finger DNA binding. *Genome Biol.* **2017**, *18*, 167.

https://doi.org/10.1021/acs.jctc.5c01690
J. Chem. Theory Comput. XXXX, XXX, XXX−XXX