https://doi.org/10.1093/nargab/lqac103

A parametric approach for molecular encodings using multilevel atomic neighborhoods applied to peptide classification

Georges Hattab *, **Aleksandar Anžel**, **Sebastian Spänig, Nils Neumann and Dominik Heider**

Department of Mathematics and Computer Science, Philipps-Universität Marburg, Marburg 35032, Germany

Received August 11, 2022; Revised November 26, 2022; Editorial Decision December 12, 2022; Accepted December 19, 2022

ABSTRACT

Exploring new ways to represent and discover organic molecules is critical to the development of new therapies. Fingerprinting algorithms are used to encode or machine-read organic molecules. Molecular encodings facilitate the computation of distance and similarity measurements to support tasks such as similarity search or virtual screening. Motivated by the ubiquity of carbon and the emerging structured patterns, we propose a parametric approach for molecular encodings using carbon-based multilevel atomic neighborhoods. It implements a walk along the carbon chain of a molecule to compute different representations of the neighborhoods in the form of a binary or numerical array that can later be exported into an image. Applied to the task of binary peptide classification, the evaluation was performed by using forty-nine encodings of twenty-nine data sets from various biomedical fields, resulting in well over 1421 machine learning models. By design, the parametric approach is domain- and task-agnostic and scopes all organic molecules including unnatural and exotic amino acids as well as cyclic peptides. Applied to peptide classification, our results point to a number of promising applications and extensions. The parametric approach was developed as a Python package (cmangoes), the source code and documentation of which can be found at https://github.com/ghattab/cmangoes and https://doi.org/10.5281/zenodo.7483771.

INTRODUCTION

Computational approaches to molecular analysis support a range of biologically oriented applications and tasks that are facilitated by the similar property principle (1). Tasks range from but are not limited to identifying the interactions between drugs and target proteins, to revealing quantitative relationships between structural properties of chemical compounds and biological activities, to screening a handful of membrane proteins for drug delivery (2–5). The similarity principle states that similar molecules will also tend to exhibit similar biophysical properties. For example, the virtual screening task is primarily used in drug discovery and allows researchers to find candidate treatments for Alzheimer's disease or HIV (6–8). Virtual screening is carried out by calculating similarity measures of compounds in a database to a reference compound. Using a similarity search, compounds are ranked in descending order and manual screening is performed on the highest ranked compounds (9). Yet to support the growing number of machine-related tasks, the structure of a molecule must be encoded to a machine-readable format. Indeed, certain structural information may be represented as a numeric feature by means of mapping a large data item to a much shorter bit string. In this context, different types of molecular fingerprints have been proposed: Substructure key-based such as MACCS (3), topological like FP2 OpenBabel (10), circular like MNA (11), pharmacophore and hybrid. This process leads to a molecular fingerprint, which uniquely identifies each molecule through data encoding.

Given such a fingerprint, we can abstract task-specific information at different levels, from the atom, to the neighborhood of an atom, to the amino acid of a protein or even to the base of a DNA molecule. Thanks to this process of abstraction, various biological and chemical aspects may be characterized, similarities and differences may be noted. In the similarity searching example, distances such as Tanimoto or Dice coefficients are calculated between the fingerprint of a certain molecule and its reference during the search (5,12). Besides previously mentioned measures, researchers have examined many other distance measures and investigated their limitations (e.g. Manhattan, Soergel) (13,14). Machine learning (ML) has been used in various domain applications (e.g. for predictions, cluster-

*To whom correspondence should be addressed. Email: georges.hattab@uni-marburg.de
Present address: Georges Hattab, Robert Koch Institut, ZKI-PH, Nordufer 20, 13353 Berlin, Germany.

ing, etc.). Different molecular properties can be used as input for training of ML models in order to achieve the best prediction performance. Different molecular properties will have different descriptive power for the source molecule. The molecular properties selected can define the similarity or dissimilarity between molecules. Our proposed approach starts from the question of whether neighborhoods are sufficiently descriptive to characterize organic molecules.

With various bioinformatics tools implementing different types of molecular fingerprints and fingerprinting algorithms (15–17), there is an immediate need for adaptable molecular approaches that could accommodate different tasks and specific user needs while respecting different domain standards. That is to say, parametric approaches where users can select and change the values of different parameters, thus adjusting the encoding method according to the, e.g. task, domain or ML model. In this work, we present a parametric approach to molecular encodings that we apply to the specific task of peptide classification. The idea is to correctly classify peptides that possess certain features. The concept of depending on the neighborhood hierarchy has not, to our knowledge and despite the existence of several fingerprinting algorithms, been considered.

To implement this concept, we depend on the element carbon (C) to produce various encodings. As the centerpiece of organic life, C is ubiquitous and very good at forming large and stable chains of various organic molecules. Inspired by its central role, we introduce a parametric approach to molecular encodings of carbon-based multilevel atomic neighborhoods as an open source standalone executable and a GitHub source repository; namely cmangoes. It takes as input positional and optional arguments allowing the creation of user-defined molecular encodings. The former include a path to one or more molecular sequences, the type of encoding (binary or discretized), and a padding parameter (centered or offset). The latter include, but are not limited to, a parameter for the upper limit of neighborhood levels to be considered, and whether or not images are required.

This parametric approach paves the way for further efforts to tailor molecular encodings to specific user requirements while taking into account the parameter space of fingerprinting algorithms. Furthermore, since its implementation follows domain-specific standards, the parametric approach can be adopted to address different tasks in a variety of domains. In the following, we introduce the methodology of the proposed parametric approach and showcase its usefulness for the example task of peptide classification via an evaluation on twenty-nine data sets and a comparison to 45 encodings in the biomedical domain.

MATERIALS AND METHODS

The presented work takes into account the ubiquity of the carbon element and its central role in holding together the structure of organic molecules and organizing their neighborhoods. The parametric approach encodes the neighborhoods around the carbon chain of a molecule in multiple levels. Various design considerations are followed to meet established domain standards and create compatible encodings for common similarity measures and distances. This section describes the parametric approach, design considerations of the underlying algorithm to fit the domain specificity of molecular fingerprinting, the data sets used, and the evaluation of the parametric approach: peptide classification and benchmark.

The parametric approach

The parametric approach handles the input data, generates intermediate data representations as a graph, traverses it to record the relevant neighborhoods according to user-specified parameters, transforms the recorded features to their final output format, and generates the corresponding representations. The walk along the carbon chain iteratively lists the neighboring atoms of each visited atom. The neighborhood of an atom is defined by the neighbors found by direct short paths around it. The hierarchies are multiple levels of an atom's neighborhood and are defined hierarchically based on their proximity to the carbon chain. By incorporating hierarchies into the encoding, molecules of varying lengths containing different substructures can be appropriately represented. An implementation of this approach is provided as a Python package for easy reproducibility. The core development of the algorithm was performed using Python programming language, version 3.8.5 (18,19). The chosen language offers high compatibility with existing computational approaches commonly used in bioinformatics and cheminformatics. All core-related dependencies are listed on the official GitHub page. The package accepts FASTA or SMILES file format specifications and follows a seven-step encoding pipeline. Figure 1 depicts an example workflow diagram for an input molecule as a SMILES string.

The first step consists of parsing and processing the input data, given in one of the two available formats. To ensure all atoms of a given molecule are present for all following steps of the parametric algorithm, hydrogen atoms (H) are added upon data import.

Second, an intermediary molecular graph data structure is employed to efficiently traverse the carbon chain and to record the relevant neighborhoods. To generate the molecular graph, the input data is parsed into an adjacency matrix which is then transformed into a graph. To create a robust and deterministic encoding, all atoms of a given molecule are represented by nodes in the molecular graph and are numbered with a unique identifier. Each node in the molecular graph stores the type of element they represent using its element symbol from the periodic table. The element labels are required in subsequent steps to generate the feature vectors. To avoid redundancy, the edges of the molecular graph do not store any additional information aside from the nodes they connect, i.e. an unweighted graph.

Third, to aid the identification of the optimal depth, the molecular graph may be visualized. When a data set is used, users select the molecule of their choice in the data set and its intermediary graph is rendered.

Fourth, the walk along the carbon chain corresponds to an iteration over a numbered list of carbon atoms. This list is created by only retaining the nodes that correspond to the carbon element symbol (C). Each carbon node is included exactly once. The filtered nodes are then sorted in ascending

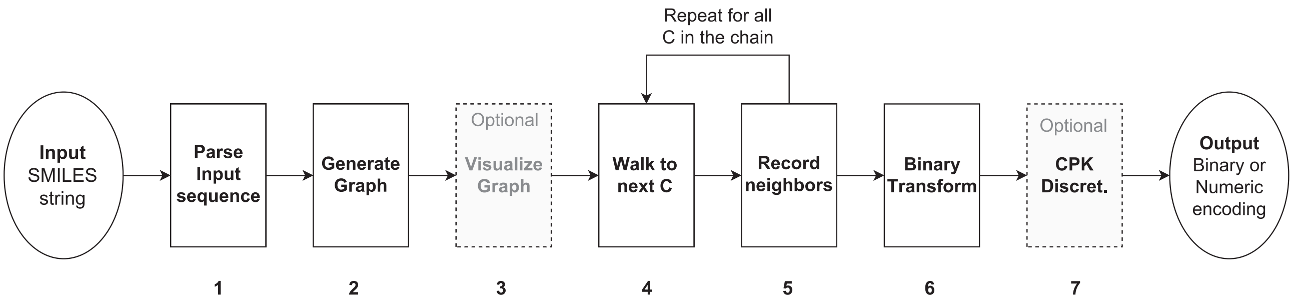


Figure 1. Example workflow of the encoding pipeline for a given molecule. C: carbon. Corey–Pauling–Koltun discretization: CPK discret.

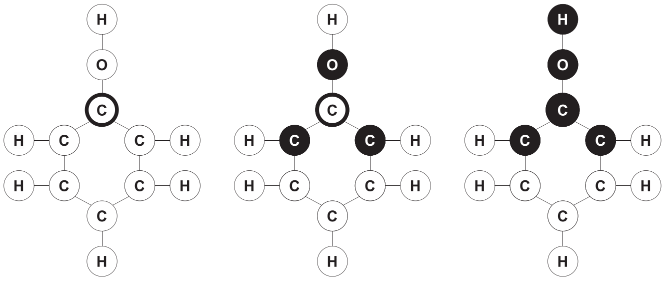
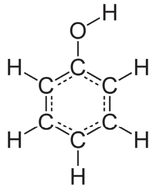


Figure 2. Visual demonstration of a computation for two-level hierarchies of the phenol molecule (C_6H_6O). Each figure corresponds to one iteration along the carbon chain. (Left) The algorithm reaches the highlighted carbon atom C at an example iteration. (Center) It records the first-level hierarchy: C, C, O. (Right) Then, the second-level hierarchy: C, H. The resulting hierarchy is C, C, O, C, H. The algorithm iterates onto the next carbon atom.

Table 1. The structural formula of the phenol molecule and its recorded neighborhoods using one- and two-level hierarchies. Phenol or C_6H_6O has the SMILES specification: `C1=CC=C(C=C1)O`. Canonical SMILES: `Oc1ccccc1`. C_0 is located at the bottom of the cycle. C_3 is at the top and is connected to the oxygen O element



C_0	C_1	C_2	C_3	C_4	C_5
C	C	C	C	C	C
C	C	C	C	C	C
H	H	H	O	H	H
C	C	C	C	C	C
-	-	-	H	-	-

order by their unique node identifier. The immediate neighbors include all nodes connected directly by an edge to the respective carbon node. Aside from the immediate neighbors, additional hierarchy levels of a neighborhood can be saved. Figure 2 shows an illustrative iteration for the example phenol molecule.

Fifth, neighborhoods along the previously mentioned walk are saved. The additional hierarchy levels are defined as the immediate neighbors of all nodes belonging to the previous level. For instance, the second-level hierarchy includes all nodes with a direct connection to any node from the first-level hierarchy. To avoid redundancy in the encoded information, an additional filter is applied when recording more than one hierarchy. Since neighborhoods are recorded as part of the main iteration, this filter excludes nodes containing carbon atoms. The number of recorded hierarchies can be set using the level parameter. The data structure used for saving the neighborhoods is a dictionary. Only the element symbols belonging to the neighborhood's nodes are saved. The list of element symbols is, by nature of the iteration, automatically sorted according to the unique node identifiers. This ensures that the feature vectors are deterministic across multiple runs of the encoding. To simplify subsequent steps of the algorithm and feature-based operations, such as transformation, the dictionary is transformed to a data frame. Table 1 reports the resulting hierarchies for the example phenol molecule.

Sixth, feature transformation (binary and discretization) is applied on the hierarchies. Feature transformation enables numeric operations and image generation of the resulting encodings. In the example of the binary encoding, the feature vectors are represented as bit strings, 0 and 1 encode the absence or presence of an atom in the respective neighborhood. The resulting categorical data frame may include missing values depending on the structure of the encoded compound. This can occur when recording more than one hierarchical level, as shown above, when carbon nodes are excluded. To preserve the integrity of the overall data structure and avoid the occurrence of an uneven number of atoms recorded in neighborhoods along the carbon chain, the data frame is automatically filled with missing values in the relevant positions. Since the value 0 represents the absence of information, this procedure does not distort the resulting feature vector.

Seventh, the numerical encodings in the image space follow either a 1-bit coding or the Corey Pauling Koltun color coding (CPK). This optional step exports images with either binary or discretized encodings (20). Tables 2 and 3 show the resulting output after feature transformation for the example phenol molecule. Figure 3 depicts the image representations of the resulting transformations.

Domain-specific standards

The created feature vectors are domain- and task-agnostic. That is to say, they are compatible with various domain-specific tasks such as database querying or virtual screening (3,5).

In the special case of cyclic molecules, for example aromatic cycles in proteins or cyclic peptides, the unique node identifiers and the sorted filtered list permit the algorithm to bypass cyclical substructures. In turn, this expands the application scope of the proposed approach to

Table 2. Recorded neighborhoods of the phenol molecule using one- and two-level hierarchies after binary transformation. To enable distance-based, similarity searching and machine learning tasks, the categorical encoding is transformed using dummy encoding

C_0C	C_0H	C_1C	C_1H	C_2C	C_2H	C_3C	C_3H	C_3O	C_4C	C_4H	C_5C	C_5H
1	0	1	0	1	0	1	0	0	1	0	1	0
1	0	1	0	1	0	1	0	0	1	0	1	0
0	1	0	1	0	1	0	0	1	0	1	0	1
1	0	1	0	1	0	1	0	0	1	0	1	0
0	0	0	0	0	0	0	1	0	0	0	0	0

Table 3. Recorded neighborhoods for the phenol molecule using one- and two-level hierarchies after CPK-based discretization. The parametric approach transforms the features to integers ranging from 0 to 16 as per the CPK coloring system

C_0C	C_0H	C_1C	C_1H	C_2C	C_2H	C_3C	C_3H	C_3O	C_4C	C_4H	C_5C	C_5H
3	0	3	0	3	0	3	0	0	3	0	3	0
3	0	3	0	3	0	3	0	0	3	0	3	0
0	2	0	2	0	2	0	0	5	0	2	0	2
3	0	3	0	3	0	3	0	0	3	0	3	0
0	0	0	0	0	0	0	2	0	0	0	0	0

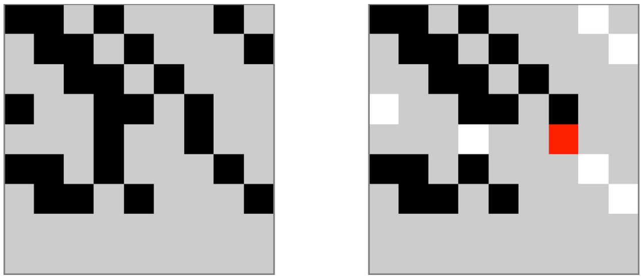


Figure 3. Image representations of the encoding for the phenol molecule. (Left) Binary encoding. (Right) CPK-color encoding. The images are created based on the feature vectors found in Tables 2 and 3, respectively.

include cyclic molecules; such as cyclic peptides often used in therapeutics (21). Table 1 reports the resulting hierarchies. Figure 2 shows an example iteration for the phenol molecule.

The image representations complement the mathematical feature vectors to provide an accessible way to understand the resulting encodings and enable additional operations in the image space (22). Figure 3 depicts the image representations of the resulting transformations for the phenol molecule. To avoid dimensional mismatches in the output feature vector and avoid bit collision for different molecule sizes, a padding step is included in the encoding pipeline when applying the encoding to more than one molecule. It includes two padding strategies to either offset (top-left shift) or center the image representation by introducing new empty pixels around the edges of an image.

Data sets

Twenty-nine data sets comprising peptides and small proteins from various biomedical domains are employed. These include immuno-modulatory and cell-penetrating peptides, but also peptides specifically targeting cancer, fungi, microbes, tuberculosis and viruses. Figure 4 lists all the data sets included in this work and reports their class imbalance or imbalance ratio for the evaluation. The properties encoded in the target vectors are represented by ones and zeros, corresponding to the presence or absence of the relevant property, respectively. For example, six data sets are cell-penetrating peptides. Used in research and medicine, they are also known as protein transduction domains and carry a variety of cargoes across the cellular membranes in an intact and functional form (23). The property encoded in the target vector is whether or not the peptide is cell-penetrating.

Peptide classification

To evaluate the parametric approach, we adopt the task of peptide classification and rely on the state-of-the-art tool PEPTIDE REACToR (16). We run a high-throughput comparison of forty-nine encodings on the aforementioned data sets (15). Based on this work, the Random Forest classifier is used with default parameters as the ML model to address the task of peptide classification. For reproducibility, the complete study details, such as hyperparameter values, data set split sizes, etc., are taken from PEPTIDE REACToR.

To ascertain whether the class-imbalance and the data set size has an effect on the prediction quality, both the class distribution of the respective target vectors and the number of observations contained in each data set vary.

To minimize bias based on the data set choice, the twenty-nine data sets are encoded with four different encodings with the parametric approach. They comprise the first- and second-level hierarchies, with a centered or shifted (offset) padding and are binary or discretized, respectively.

The evaluation is carried out by adding the four encodings to the aforementioned tool. This totals forty-nine encodings. The effective comparison of the classification results relies on the F_β Score metric with $\beta = 1$. It corresponds to the weighted harmonic mean of precision and recall, reaching its optimal value at 1 and its worst value at 0.

The evaluation of the peptide classification task comprises training 1421 ML models which result from forty-nine encodings applied to 29 data sets. The evaluation was carried out using cloud computing. We relied on the de.NBI

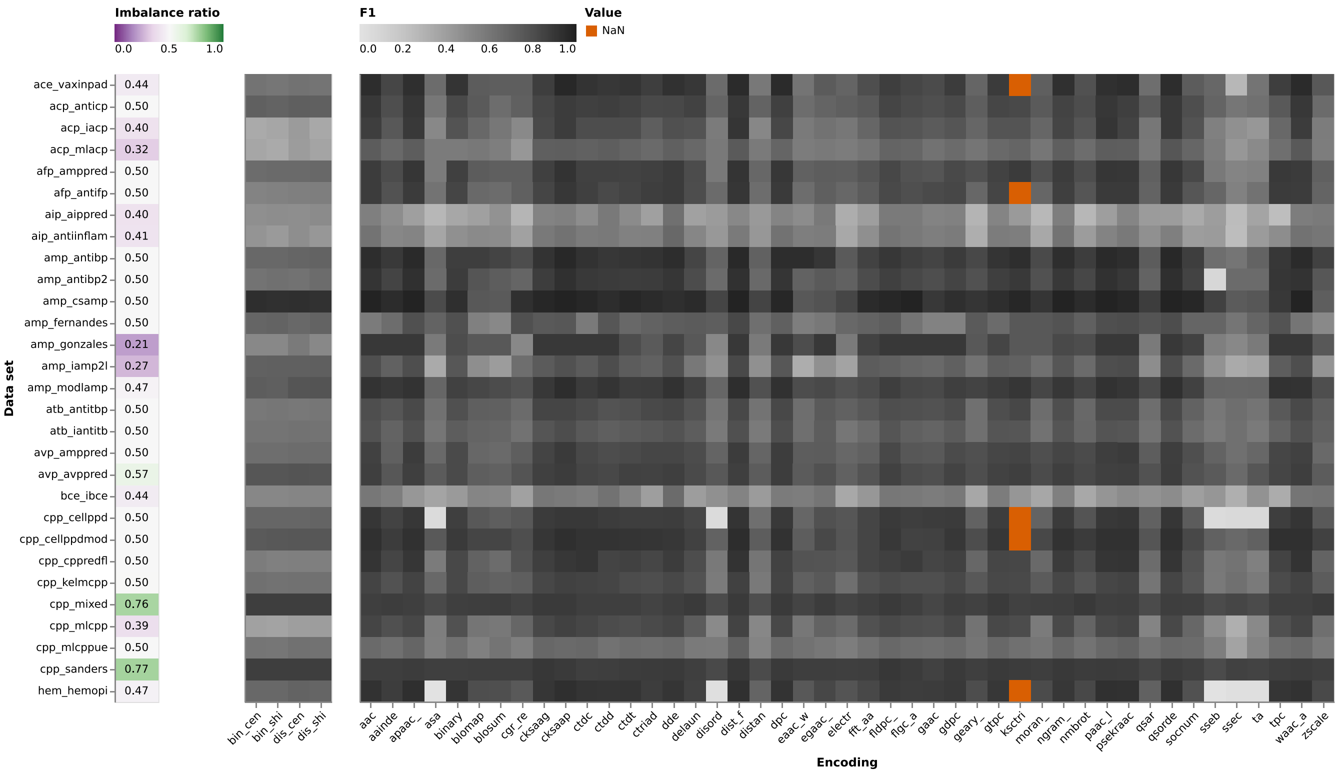


Figure 4. Evaluation results of the peptide classification task. The four encodings obtained using the parametric approach are shown on the left versus the 45 sequence- and structure-based encodings. Results are sorted by class imbalance and encoding type. Color coding corresponds to the maximum F_1 Score of the bootstrapped medians for a group. The abscissa is organized by sequence- and structure-based encodings. The ordinate is sorted by class imbalance (cut-off 0.35). Groups are separated by white bars.

Cloud within the German Network for Bioinformatics Infrastructure.

Benchmark

To report the performance results of the proposed parameter approach, it is benchmarked as a fingerprinting algorithm. We consider two parameters for benchmarking the creation of an encoding: the elapsed time in seconds and the amount of encoded data in bytes. Benchmarking is performed for all four encodings on all data sets. For each encoding, six runs are performed and benchmarked.

Benchmarking is conducted using multi-threading on a Linux machine. Kernel: 5.17.5-76051705-generic, CPU: Intel i7-10700 (16) @ 2.90GHz (Turbo 4.90GHz), Thread(s) per core: 2, Core(s) per socket: 8, Memory: 16GB.

RESULTS

The parametric approach provided a simplified set of parameters to adapt the encoding step to user-specific needs. It is available as a standalone Linux executable and the source code GitHub repository to create encodings, explore their parameter space, and generate hypotheses and design ML experiments.

By relying on the F_1 score, we found that the four encodings were consistently providing equivalent results with marginal differences. By conducting a one-way ANOVA test, we did not find statistically significant differences among the four encodings. At $P < 0.05$ and three degrees of freedom (df) between-groups and 112 df within-groups, the F-statistic value was 0.084 and the P-value was 0.969. While the F-statistic informed us whether there is an overall difference between the sample mean, the Tukey's range test or Tukey HSD allowed us to determine that there is no significant difference between the various pairs of means. In other words, we found that there is no significant difference in performance if the user chooses a binary or discretized encoding type, and in the padding strategy (center or shifted) for the peptide classification task. Results of the Tukey HSD are reported in Table 4.

The evaluation of the classification task showed that the first- and second-level hierarchies carry enough information to reach acceptable and good classification results. However, the results are fairly sparse across the different data sets and follows the general trend of other existing encodings. A complete overview of the evaluation results using the F_1 score are reported in Figure 4. A Jupyter Notebook (`Code/visualize.ipynb`) is made available on GitHub to reproduce all figures and provide interactive visualizations.

The benchmarking of the parametric approach permitted us to report its performance on different data sets. Benchmarking results are visualized and complemented with the imbalance ratio of the twenty-nine data sets in Figure 5. Overall, the elapsed time (s) shows a linear dependency with the data set size (bytes). Our results indicate that a data size

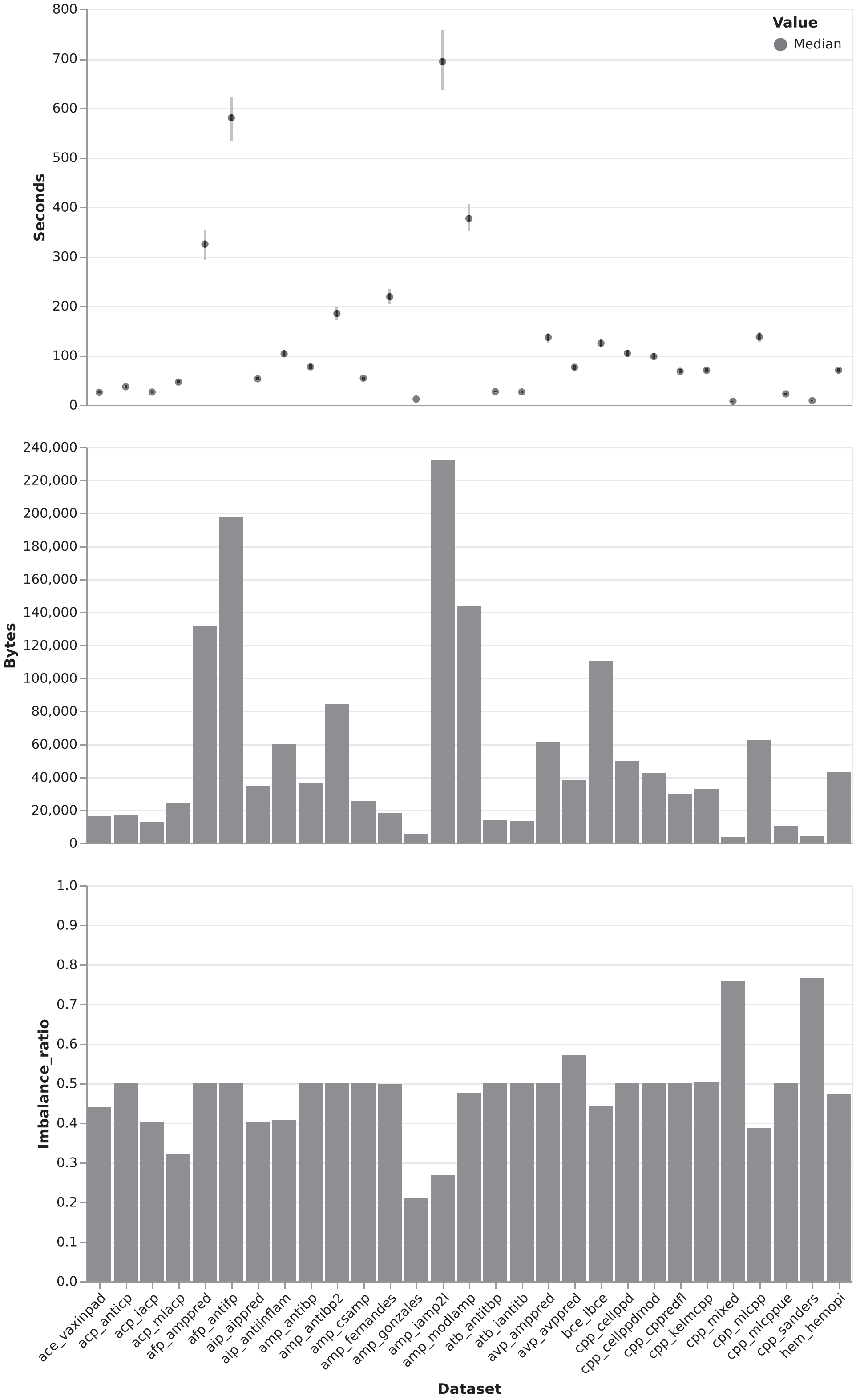


Figure 5. Benchmark performance of the four encodings created using the parametric approach. Median values are reported and were calculated for all six runs and encodings. Faceted views of the elapsed time (s), the size of the encoded data (byte), and the imbalance ratio of the data set.

Table 4. Tukey's range test results. E1, E2, E3, E4 are the four encodings `bin_cen`, `bin_shi`, `dis_cen`, `dis_shi`, respectively. M1 to M4 are the means of each encoding group results of the F_1 score. Q refers to the fact that Tukey's range test is based on a studentized range distribution (q)

Pairwise comparisons		$\mathrm{HSD}_{0.05} = 0.098$ $\mathrm{HSD}_{0.01} = 0.119$	$Q_{0.05} = 3.688$ $Q_{0.01} = 4.504$

E1:E2	M1 = 0.62
	M2 = 0.62
		0.00	$Q = 0.08$ ($P = 0.99993$)

E1:E3	M1 = 0.62
	M3 = 0.63
		0.01	$Q = 0.52$ ($P = 0.98262$)

E1:E4	M1 = 0.62
	M4 = 0.63
		0.01	$Q = 0.36$ ($P = 0.99430$)

E2:E3	M2 = 0.62
	M3 = 0.63
		0.02	$Q = 0.61$ ($P = 0.97347$)

E2:E4	M2 = 0.62
	M4 = 0.63
		0.01	$Q = 0.44$ ($P = 0.98948$)

E3:E4	M3 = 0.63
	M4 = 0.63
		0.00	$Q = 0.17$ ($P = 0.99942$)

of 1 MB required 2245 median seconds. That is to say, the encoding of 1 byte requires 2.141 median ms.

DISCUSSION

First, further computational improvements can be made for both the parametric approach and the evaluation pipeline. On one hand, very large data sets can be batched encoded and as such parallel processing of the input molecules is relevant. On the other hand, the large number of training iterations makes computational optimizations especially important. Although we rely on two-dimensional multilevel neighborhoods alone, this work provides a proof of concept and the evaluation of other fingerprinting algorithms for binary classification should be considered, as reported in previous work (16). Both sequence and structure encodings were included in the evaluation. Indeed, our results can be directly compared to the classification results reported in the `PEPTIDE REACToR` tool. We hope this effort enables a fair and direct comparison across encodings and data sets.

Second, compared to results reported in the related work, our results were found to be consistent which made the parametric approach a dependable one. Results using the F_1 score showed an acceptable to good separation of the two classes, i.e. robustness. In the example of the six cell-penetrating peptides data sets (*cpp*), it is important to note that in the majority of the original works, both the accuracy and the Matthews Correlation Coefficient (MCC) performance metrics were used and this is in discordance with good practices for binary classification. The AUC usually provides robustness of the resulting classifier and is more discriminative than the MCC, while the accuracy is the measure of the closeness to a specific value and the AUC is the measure across all the possible thresholds (24–26). We chose the F_1 score because it is applicable to any particular point on the ROC curve. While the AUC is the area under the ROC curve, the F_1 score is a measure of precision and recall at a particular threshold value. To maximize this score, both precision and recall must be high. In this ideal case, the model returns many results, all correctly labeled.

Third, we found no significant difference in performance if the user chooses a binary or discretized encoding type as well as in the padding strategy (center or shifted). This may imply that the strongest signal comes from the main positional parameter: the levels to be considered. Although other explanations are possible such as the domain of peptides or the length of a molecule, results portray the robustness of the parametric approach. In this case, the different possible combinations of the feature vector representation (via the optional parameters) did not affect the classification results. In addition, we have discovered that employing only the first or second level leads to subpar performance, albeit this is not covered in this study. In fact, this point has not been addressed because looking at just one level contradicts the logic behind the parametric approach methodology.

Fourth, the image representation of the resulting encodings constitutes an interesting research starting point. It opens up a new space of representation by using the image domain. For example, convolutional neural networks may be used for the same task of classification yet by relying on the images of the resulting encodings. Since such neural networks convolve learned features with input data, and use two-dimensional convolutional layers, their architecture is suited to processing two-dimensional data, such as images. Such methodology would eliminate the need for manual feature extraction required to classify the images.

Fifth, the molecular complexity field is noteworthy. It provides fundamental concepts that underly current fragment-based lead discovery. It considers the general index of molecular complexity, where features that make a molecule more or less complex are taken into account (27,28). For example, size, symmetry, branching, rings, multiple bonds and heterogeneity in the atoms. Such concepts have been used in various application domains such as chromatography analysis and synthesis pathways. It would be very useful to rely on such features to improve the proposed approach and introduce further parameters such as symmetry, the presence of a cycle, or even the distances among atoms. Such additions may be made at the second step of the parametric approach to enrich the resulting encodings, increase the user-settable parameters, and further vary the resulting performance of an encoding for a specific task or domain.

Sixth, although this parametric approach proved useful for cell-penetrating peptides and achieved acceptable classification results for different data sets, it is important to extend its usage to include larger molecules and more heterogeneous data sets such as membrane proteins (29,30). For comparability, we successfully evaluated additional data sets, including imbalanced and large data sets that broadened the application scope. Furthermore, it would be valuable to consider correlation results among varying encodings. This could open up the way to build upon the parametric approach and bypass computationally demanding algorithms and move directly to the design of ML experiments.

Seventh, by default the parametric approach produces very sparse encodings. This is especially the case when the encodings are padded or centered. Hence, it is important to develop specialized methods to address sparsity and evaluate its effects. This relates to the problem of representation and has potential links to data compression. Further considerations are warranted for a more faithful space of representation so to reduce the data and preserve its relevant structure.

Eighth, this work started with the question of whether atomic neighborhoods are descriptive enough to characterize organic molecules. Although this is a naive question, it is related to the basic idea that the neighborhoods created by the carbon atom are not only important but may be sufficient to obtain good classification results. To potentially achieve very good or perfect classification results, the parametric approach can be complemented by the molecular complexity concepts mentioned above. Moreover, the first version of the parametric approach cannot handle other atoms than the carbon atom as the backbone of a molecule. However, heterocyclic compounds can be encoded and the bonds between the different atoms are respected.

Ninth, although the parametric approach is focused on organic compounds or molecules, it is possible to adapt the underlying algorithm to create multilevel atomic neighborhoods of molecules that lack C–H bonds. That is, considering inorganic polymers whose backbone structure does not include carbon atoms, further expanding the application domains and tasks for which the parametric approach could be used.

Tenth and last, evaluating all models using good practices in ML is a standard approach to optimize the prediction performance of the models. Since the parametric approach provides a parameter space, researchers may also move upstream and consider a sensitivity analysis to better fine-tune resulting ML models. Moreover, geometrical deep learning tools, like graph neural networks (GNNs), could be incorporated to further improve the overall ML aspect. This method would exploit the underlying machine representation of molecules using graphs (i.e. adjacency matrices). Work in this direction is already underway and can be seen in (31,32).

CONCLUSION

The presented parametric approach is created as an easy-to-use and easy-to-install solution that includes the necessary operations to create custom multilevel encodings of molecular data. Results for the binary peptide classification task were produced by using the PEPTIDE REACToR tool. The F_1 score reached 0.86 even with a class imbalance of 0.76 and 0.77. The best performance reached 0.93 for the antimicrobial activity prediction in Cysteine-Stabilized peptides (data set: *amp_csamp*) (33). Moreover, the performance evaluation showed that the first two-level hierarchies carry the most meaningful information for the classification task. Overall, the classification results of the four encodings were consistent with and comparable to the general trend of the state-of-the-art results. Benchmark results indicated that the parametric approach is not computationally intensive and linearly increases with the data set size. Since fingerprint representations decrease computational expenses and enable rapid comparison of different molecules, future work could extend the application of this approach beyond the task of binary classification and peptides. Unlike other fingerprinting algorithms and methods, the intermediate graph data structure makes the parametric approach versatile and permits the usage of organic molecules such as unnatural and exotic amino acids and cyclic peptides. Moreover, we foresee that the proposed work will be a valuable tool to complement and enhance current molecular fingerprinting algorithms and offer further insights into the parameters and the use of hierarchies and their potential combination.

DATA AVAILABILITY

The data underlying this article are available in Zenodo, at https://doi.org/10.5281/zenodo.7483771.

ACKNOWLEDGEMENTS

This work was supported by the BMBF-funded de.NBI Cloud within the German Network for Bioinformatics Infrastructure (de.NBI) (031A532B, 031A533A, 031A533B, 031A534A, 031A535A, 031A537A, 031A537B, 031A537C, 031A537D, 031A538A).

Author contributions: Conceptualization, G.H. and N.N.; methodology, G.H., N.N. and A.A.; software, N.N., A.A. and S.S.; validation, G.H. and A.A.; investigation, G.H. and A.A.; resources, A.A. and S.S.; data curation, G.H., A.A and S.S.; writing—original draft preparation, G.H.; writing—review and editing, G.H., A.A. and D.H.; visualization, G.H., N.N. and A.A.; supervision, D.H.; project administration, G.H.; funding acquisition, D.H. All authors have read and agreed to the published version of the manuscript.

FUNDING

This research was funded by the Hessian Ministry for Science and the Arts (LOEWE) in the context of the Molecular Storage for Long-Term Archiving (MOSLA) consortium.

Conflict of interest statement. The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

REFERENCES

1. Johnson,M.A. and Maggiora,G.M. (1990) In: *Concepts and Applications of Molecular Similarity*. Wiley.
2. Csermely,P., Korcsmáros,T., Kiss,H.J., London,G. and Nussinov,R. (2013) Structure and dynamics of molecular networks: a novel paradigm of drug discovery: a comprehensive review. *Pharmacol. Ther.*, **138**, 333–408.
3. Cereto-Massagué,A., Ojeda,M.J., Valls,C., Mulero,M., Garcia-Vallvé,S. and Pujadas,G. (2015) Molecular fingerprint similarity search in virtual screening. *Methods*, **71**, 58–63.
4. Neves,B.J., Braga,R.C., Melo-Filho,C.C., Moreira-Filho,J.T., Muratov,E.N. and Andrade,C.H. (2018) QSAR-based virtual screening: advances and applications in drug discovery. *Front. Pharmacol.*, **9**, 1275.
5. Bajusz,D., Rácz,A. and Héberger,K. (2017) Chemical data formats, fingerprints, and other molecular descriptions for database analysis and searching. In: *In Silico Drug Discovery Tools*. Elsevier Inc., pp. 329–378.
6. Ponzoni,I., Sebastián-Pérez,V., Martínez,M.J., Roca,C., De la Cruz Pérez,C., Cravero,F., Vazquez,G.E., Páez,J.A., Díaz,M.F. and Campillo,N.E. (2019) QSAR classification models for predicting the activity of inhibitors of beta-secretase (BACE1) associated with Alzheimer's disease. *Sci. Rep.*, **9**, 9102.

7. Vora,J., Patel,S., Sinha,S., Sharma,S., Srivastava,A., Chhabria,M. and Shrivastava,N. (2019) Molecular docking, QSAR and ADMET based mining of natural compounds against prime targets of HIV. *J Biom. Struct. Dyn.*, **37**, 131–146.
8. Dybowski,N.J., Riemenschneider,M., Hauke,S., Pyka,M., Verheyen,J., Hoffmann,D. and Heider,D. (2011) Improved Bevirimat resistance prediction by combination of structural and sequence-based classifiers. *BioData Min.*, **4**, 26.
9. Willett,P. (2006) Similarity-based virtual screening using 2D fingerprints. *Drug Disc. Today*, **11**, 1046–1053.
10. O'Boyle,N.M., Banck,M., James,C.A., Morley,C., Vandermeersch,T. and Hutchison,G.R. (2011) Open Babel: An open chemical toolbox. *J. Cheminformatics*, **3**, 33.
11. Filimonov,D., Poroikov,V., Borodina,Y. and Gloriozova,T. (1999) Chemical similarity assessment through multilevel neighborhoods of atoms: definition and comparison with the other descriptors. *J. Chem. Inf. Comput. Sci.*, **39**, 666–670.
12. Deepak,P. and Deshpande,P.M. (2015) In: *Operators for Similarity Search: Semantics, Techniques and Usage Scenarios*. Springer.
13. Riniker,S. and Landrum,G.A. (2013) Similarity maps-a visualization strategy for molecular fingerprints and machine-learning methods. *J. Cheminformatics*, **5**, 43.
14. Godden,J.W., Xue,L. and Bajorath,J. (2000) Combinatorial preferences affect molecular similarity/diversity calculations using binary fingerprints and Tanimoto coefficients. *J. Chem. Inf. Comput. Sci.*, **40**, 163–166.
15. Spänig,S. and Heider,D. (2019) Encodings and models for antimicrobial peptide classification for multi-resistant pathogens. *BioData Min.*, **12**, 7.
16. Spänig,S., Mohsen,S., Hattab,G., Hauschild,A.-C. and Heider,D. (2021) A large-scale comparative study on peptide encodings for biomedical classification. *NAR Genom. Bioinform.*, **3**, lqab039.
17. Sequeira,A.M., Lousa,D. and Rocha,M. (2022) ProPythia: a Python package for protein classification based on machine and deep learning. *Neurocomputing*, **484**, 172–182.
18. Van Rossum,G. and Drake,F.L. (2009) In: *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA.
19. Oliphant,T.E. (2007) Python for scientific computing. *Comput. Sci. Eng.*, **9**, 10–20.
20. Hattab,G., Rhyne,T.-M. and Heider,D. (2020) Ten simple rules to colorize biological data visualization.*PLoS Comput. Biol.*, **16**, e1008259.
21. Sugita,M., Sugiyama,S., Fujie,T., Yoshikawa,Y., Yanagisawa,K., Ohue,M. and Akiyama,Y. (2021) Large-scale membrane permeability prediction of cyclic peptides crossing a lipid bilayer based on enhanced sampling molecular dynamics simulations. *J. Chem. Inf. Model.*, **61**, 3681–3695.
22. Keim,D.A., Mansmann,F., Schneidewind,J. and Ziegler,H. (2006) Challenges in visual data analysis. In: *Tenth International Conference on Information Visualisation (IV'06)*. IEEE, pp. 9–16.
23. Taylor,R.E. and Zahid,M. (2020) Cell penetrating peptides, novel vectors for gene therapy. *Pharmaceutics*, **12**, 225.
24. Ling,C.X., Huang,J. and Zhang,H. (2003) AUC: A statistically consistent and more discriminating measure than accuracy. In: *IJCAI*. Vol. **3**, pp. 519–524.
25. Calders,T. and Jaroszewicz,S. (2007) Efficient AUC optimization for classification. In: *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, pp. 42–53.
26. Halimu,C., Kasem,A. and Newaz,S.S. (2019) Empirical comparison of area under ROC curve (AUC) and Mathew correlation coefficient (MCC) for evaluating machine learning algorithms on imbalanced datasets for binary classification. In: *Proceedings of the 3rd International Conference on Machine Learning and Soft Computing*. pp. 1–6.
27. D'Amboise,M. and Bertrand,M.J. (1986) General index of molecular complexity and chromatographic retention data. *J. Chromatogr. A*, **361**, 13–24.
28. Hendrickson,J.B., Huang,P. and Toczko,A.G. (1987) Molecular complexity: a simplified formula adapted to individual atoms. *J. Chem. Inf. Comput. Sci.*, **27**, 63–67.
29. Chou,K.-C. and Elrod,D.W. (1999) Prediction of membrane protein types and subcellular locations. *Proteins Struct. Func. Bioinform.*, **34**, 137–153.
30. Hattab,G., Warschawski,D.E., Moncoq,K. and Miroux,B. (2015) Escherichia coli as host for membrane protein structure determination: a global analysis. *Sci. Rep.*, **5**, 12097.
31. Wieder,O., Kohlbacher,S., Kuenemann,M., Garon,A., Ducrot,P., Seidel,T. and Langer,T. (2020) A compact review of molecular property prediction with graph neural networks. *Drug Disc. Today: Technol.*, **37**, 1–12.
32. Gasteiger,J., Becker,F. and Günnemann,S. (2021) GemNet: universal directional graph neural networks for molecules. In: Ranzato,M., Beygelzimer,A., Dauphin,Y., Liang,P. and Vaughan,J.W. (eds). *Advances in Neural Information Processing Systems*, Vol. **34**. Curran Associates, Inc., pp. 6790–6802.
33. Porto,W.F., Pires,Á.S. and Franco,O.L. (2012) CS-AMPPred: an updated SVM model for antimicrobial activity prediction in cysteine-stabilized peptides. *PLoS One*, **7**, e51444.