MULTITASK CLASSIFICATION OF ANTIMICROBIAL PEPTIDES FOR SIMULTANEOUS ASSESSMENT OF ANTIMICROBIAL PROPERTY AND STRUCTURAL FOLD

Michaela Areti Zervou [1,2] *Effrosyni Doutsi* [2] *Yannis Pantazis* [3] *Panagiotis Tsakalides*[1,2]

[1] University of Crete, Department of Computer Science, Heraklion, Greece
[2] Institute of Computer Science-FORTH, Heraklion, Greece
[3] Institute of Applied and Computational Mathematics-FORTH, Heraklion, Greece

ABSTRACT

Antimicrobial peptides (AMPs) play a significant role in guiding drug design, advancing targeted therapies, and cancer treatment research. The function of peptides is highly associated with their three-dimensional structure. AMPs particularly favor alpha-helical structures, or alpha-folds, due to their ability to disrupt the protective layers that surround cells effectively and their structural stability. Existing classifiers mainly identify AMPs but overlook their structural fold which can provide valuable insights into their function. To address this limitation, we introduce an innovative multitask classifier that recognizes AMPs and predicts their alpha-helical folds simultaneously. Our approach employs k-mers and Transformer networks for efficient, accurate multitask classification. Results on the datasets indicate comparable performance compared to single-task methods in half the time and complexity.

Index Terms— Antimicrobial peptides, Multitask classification, Transformers, k-mers.

1. INTRODUCTION

Antimicrobial Peptides (AMPs) play a pivotal role in the innate immune system of various organisms, demonstrating a high level of activity against a broad spectrum of pathogens [1]. Accurate classification of AMPs is essential for understanding their biological role and possible therapeutic uses such as in drug discovery [2]. An important aspect that determines the biological function of a protein is its three-dimensional structure which is decided by its amino acid sequence via the process of protein folding [3]. Structural class prediction provides a better understanding of the protein's biological activity and it is helpful for analyzing protein function, interactions, and regulation [4]. In the case of AMPs, the specific folding pattern can influence their interaction with bacterial membranes, modulating their ability to disrupt microbial integrity [5]. Alpha-helical structures, or alpha-folds, have been particularly favored among AMPs due to their membrane-penetrating properties and remarkable stability [6]. The amphipathic nature of alpha helices allows them to interfere with and damage the protective layers that surround cells, exerting antimicrobial effects [7]. Consequently, discerning the presence of an alpha-helical fold within an AMP sequence can offer critical insights into its potential antimicrobial activity.

State-of-the-art AMP classifiers [8, 9, 10, 11] primarily focus on AMP property detection, overlooking the structural fold's valuable insights into AMP function. These classifiers rely on a multitude of biological attributes for accurate classification resulting in increased computational complexity, potentially hindering efficiency and scalability. To address these limitations, this work proposes a novel multitask classification approach that efficiently assesses two critical aspects simultaneously: (i) the presence of the antimicrobial property within a given sequence, and (ii) whether the sequence exhibits an alpha-helical fold. Multitask classification addresses both tasks concurrently capitalizing on synergies, regularization, and generalization that arise from learning shared representations. The proposed multitask classifier exploits solely the k-mers representation [12] and Transformer networks [13]. The k-mers method captures local structural and functional properties by representing biological sequences as subsequences of length k [14], while the Transformer network captures long-range interactions across the entire sequence. Experimental evaluation on real protein data highlights the effectiveness of the proposed model by showcasing reduced training times without sacrificing significantly individual task performance. To the best of our knowledge, this is the first work that introduces multitask classification of both AMP property and structural fold, while also presenting a classifier that does not rely on biological information.

The key contributions of this paper are the following: (i) It introduces an innovative multitask classifier that relies solely on k-mers information and Transformer networks to identify simultaneously the AMP property and structural fold of a given sequence. (ii) Regardless of the dataset size, it results in

This work was funded by the TITAN ERA Chair project (contract no. 101086741) within the Horizon Europe Framework Program of the European Commission and the Hellenic Foundation for Research and Innovation (HFRI) Ph.D. Fellowship grant no. 5647.

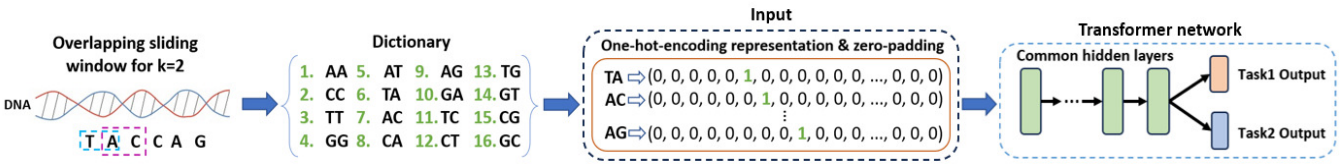

Fig. 1. An example of the proposed DNA sequence preprocessing and classification for $k = 2$.

a 50% time reduction both in training and testing, maintaining individual task performance. (iii) It demonstrates stable and consistent performance, enhancing its reliability for real-world applications.

2. MATERIALS AND METHODS

2.1. Datasets Description

To validate the classification performance of the proposed classifier in both single-task and multitask scenarios, this study employs two distinct AMP datasets from the works of Yan *et al.* [11] and Gupta and Zou [6]. Dataset 1 (Yan *et al.* [11]) contains 1,500 AMPs and an equal number of non-AMPs, each spanning 40 to 100 amino acids. Dataset 2 (Gupta and Zou [6]) consists of 5,200 protein sequences, including 2,600 established antimicrobial peptides. These sequences have amino acid lengths spanning from 10 to 50, notably shorter than those in the first dataset. Before training, the protein sequences are transformed into cDNA. Then, to enable multitask classification, the Porter 5 tool [15] is used to convert each protein sequence into a sequence of secondary structure elements: helix (H), strand (E), and coil (C) while preserving the original sequence length. Following Gupta and Zou's approach, sequences with over 10 consecutive H elements are deemed likely to fold into alpha-helical structures. By integrating this structural information with existing antimicrobial peptide annotations, we construct a multitask classification dataset capturing both antimicrobial activity and potential structural traits of the peptides.

2.2. Data Representation via k-mers

The k-mers method is a widely recognized bioinformatics tool employed in various applications [12]. In this study, DNA sequences are encoded using k-mers to align with the structure of transformer networks. The concept of k-mers involves consecutive subsequences with a length of k. Given a DNA sequence of length L, the count of k-mers in the sequence equals $L - k + 1$, and the overall potential count of k-mers for the four DNA nucleotides (A, T, G, and C) amounts to 4^k. Consequently, the number of tokens of the Transformer is 4^k and increases exponentially with k. Larger values of k amplify the input dimension (4^k $\times$ number of samples $\times$ maximum sequence length), leading to an increased model size due to a greater number of parameters. Due to the Transformer's capability to capture detailed sequence information, we employ a tokenization process with $k = 2$ aiding in computational efficiency without sacrificing the ability to capture essential details.

Sequence processing and representation: The process of identifying k-mers employs an *overlapping sliding window* technique (Fig. 1). In this approach, a window of constant length k traverses the sequence, shifting by one position at a time. This operation is reiterated for all positions within the sequence, culminating in an assemblage of k-mers. Subsequently, a dictionary is constructed, encompassing all conceivable k-mers derived from the input sequences. This dictionary encompasses 4^k distinct k-mers, with each being assigned a unique index, which subsequently facilitates one-hot-encoding. Thereafter, the categorical data are encoded as binary vectors utilizing the one-hot-encoding technique. Each k-mer becomes a binary vector of length 1×4^k, assigning 1 at the corresponding dictionary index and 0 elsewhere. Finally, to ensure consistent sequence lengths, one-hot encoding is zero-padded to match the longest sequence length ($L_{max} - k + 1$). This involves appending zero vectors of size 1×4^k to the input. The process standardizes sequence lengths, enabling compatibility across diverse machine-learning models.

3. MULTITASK TRANSFORMER CLASSIFIER

This section presents the proposed multitask Transformer classifier for sequential data. It utilizes Transformer-based encoders to capture intricate patterns in input sequences and handles multiple classification tasks simultaneously. The code is available on `https://github.com/aretiz/amp-multitask-classifier/`.

Positional Encoding: The model incorporates positional encoding to handle sequential input data. This enriches input embeddings with positional details, enabling the model to understand element order and relationships. For each token at position i in a sequence, a sine function generates the positional encoding for dimensions indexed by even values of j while a cosine function generates it for dimensions indexed by odd values of j.

Encoder Architecture: The model employs a transformer encoder with single layers featuring multi-head self-attention and feedforward neural networks. Self-attention computes weighted sums based on query, key, and value similarities. To capture diverse relationships, multi-head attention is employed which is followed by a position-wise

	Learning rate	Batch size	Hidden states	Attention heads	Transformer layers	Feedforward layer dimension	Dropout
Dataset 1	0.001	64	32	2	4	64	0.1
Dataset 2	0.001	8	64	4	2	32	0.2

Table 1. Network parameters employed in this work for Dataset 1 and Dataset 2.

feedforward neural network to further process the representations. This network consists of linear layers, with a ReLU activation function in between.

Combined Loss Function: For multi-task learning, a combined loss function is employed to simultaneously optimize predictions for two classification tasks. The loss function, given by,

$$\mathcal{L}_{\text{combined}} = \lambda \cdot \mathcal{L}_1 + (1 - \lambda) \cdot \mathcal{L}_2, \quad (1)$$

where $\mathcal{L}_1$ and $\mathcal{L}_2$ represent the loss calculated for the first and the second classification task respectively, and λ is a hyperparameter that controls the balance between the two losses. The λ values range between 0 and 1, where 0.5 would indicate an equal emphasis on both tasks. The weight parameter λ modulates the emphasis on each task, allowing the model to balance its performance across both tasks.

Multitask Prediction: After the transformer encoder processes the input, contextual information flows into task-specific linear layers called classification heads. These classification heads play a role analogous to linear transformations, as they transform the contextualized representations into meaningful class predictions for their respective tasks.

4. EXPERIMENTAL SETUP

The architecture of the proposed model was developed in PyTorch on a desktop computer equipped with NVIDIA's GPU model Quadro P4000 which demonstrates the practical feasibility of our approach. To ensure robustness, we conducted extensive experimentation and fine-tuned the hyperparameters to maximize the model's accuracy separately on Dataset 1 and Dataset 2. Specifically, a grid search method was used to determine the optimal hyperparameter set. Each hyperparameter set underwent 5-fold cross-validation, and the one with the highest average accuracy was selected. Specifically, the optimal parameters for Datasets 1 and 2 are presented in Table 1 and are utilized in both single-task and multitask architectures. The optimization process utilized the Adam optimizer with a constant learning rate, along with exponential decay rates for the first and second moment estimates set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$, respectively. Categorical cross entropy and categorical accuracy were employed as loss function and performance measures during the training process for both tasks, respectively. Additionally, an early stopping criterion was used to terminate the training process when the validation loss did not decrease for 40 consecutive epochs. Model performance is based on the lowest validation loss during training. For each dataset, the data are stratified into non-overlapping sets: 60% for training, 20% for validation, and 20% for testing. Classification is repeated 10 times with random data splits.

Finally, we employ a multi-objective evaluation approach, based on the estimation of the Pareto front [16]. This approach considers multiple performance metrics, including classification accuracy, F1-score, and area under the ROC curve (AUC), as our objectives, to identify a set of non-dominated solutions or model configurations. Formally, for two models A and B, A is said to dominate B (A $<$ B) if and only if A is not worse than B in all metrics ($A_i \leq B_i$ for all i) and A is strictly better than B in at least one metric ($A_j < B_j$ for at least one j).

5. EVALUATION RESULTS

5.1. Multitask Classification

As the shared training procedure is governed by the λ parameter, which dictates the weight assigned to each task in the loss function, we analyze the classifier's behavior across different λ values. Figure 2, illustrates the classifiers' performance for various λ values ranging from 0.1 to 0.9 for both Dataset 1 (above) and Dataset 2 (below). Notably, this figure determines a Pareto front, where each point corresponds to a specific λ value and the mean classification accuracy for each task. The observed differences in the optimal λ range between Dataset 1 and Dataset 2 are noteworthy. In Dataset 1, the model's performance reaches its peak in the range of 0.2 to 0.4 on the Pareto front. This suggests that for Dataset 1, the contributions from Task 2 (alpha-fold prediction) should not dominate the loss function. In contrast, Dataset 2 exhibits an optimal λ range between 0.3 and 0.7 on the Pareto front, indicating that for this dataset, a more balanced contribution between both tasks is beneficial. Additionally, Table 2 depicts the average performance and standard deviation of the multitask classifier for Dataset 1 and 2, for different λ values spanning from 0.1 to 0.9. The results demonstrate that the classifier achieves high AUC and F1 scores for every λ value on the Pareto front in both datasets. This highlights the multitask classifier's ability to effectively manage true positives, false positives, and false negatives which is essential in real-world applications where misclassifications can have significant consequences. The classifier's capacity to provide reliable predictions across a spectrum of λ values suggests its robustness in handling diverse application scenarios.

5.2. Comparison with Single-task Classification

The performance of the multitask classifier on the single-task scenarios that correspond to the extreme λ values, λ=1 and

Dataset 1	λ=0.2		λ=0.3		λ=0.4		λ=0.5		λ=1	λ=0
	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2
Accuracy	85.6 (2.2)	86.1 (1.0)	86.6 (1.4)	85.8 (0.3)	86.5 (1.1)	85.2 (1.4)	85.5 (1.3)	84.2 (0.9)	86.9 (1.2)	85.6 (0.8)
F1-score	85.7 (2.0)	87.3 (0.9)	86.6 (1.4)	87.2 (0.3)	86.8 (1.2)	86.9 (1.1)	85.3 (1.6)	85.6 (0.9)	87.1 (1.1)	87.1 (0.7)
AUC	93.7 (1.1)	93.5 (0.6)	94.3 (0.9)	93.1 (0.2)	94.7 (0.6)	93.2 (0.9)	94.3 (0.8)	92.0 (0.4)	94.4 (0.8)	93.3 (0.4)

Dataset 2	λ=0.3		λ=0.4		λ=0.5		λ=0.7		λ=1	λ=0
	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2	Task 1	Task 2
Accuracy	77.0 (2.7)	75.9 (1.4)	78.9 (0.8)	75.4 (1.5)	80.1 (2.3)	75.3 (1.7)	79.9 (2.1)	75.9 (0.4)	81.1 (1.4)	74.7 (1.2)
F1-score	76.5 (2.9)	81.6 (1.1)	77.3 (1.3)	81.1 (1.0)	79.4 (2.5)	81.3 (2.0)	79.0 (3.5)	82.0 (0.5)	80.9 (1.7)	80.2 (1.1)
AUC	84.6 (3.1)	81.8 (1.7)	86.9 (1.0)	82.7 (1.2)	88.0 (1.7)	82.4 (0.3)	88.3 (0.8)	82.0 (1.7)	88.2 (0.6)	80.7 (1.3)

Table 2. Effect of λ values in multitask classification of AMP (Task 1) and alpha-fold (Task 2). Standard deviation is reported in parentheses.

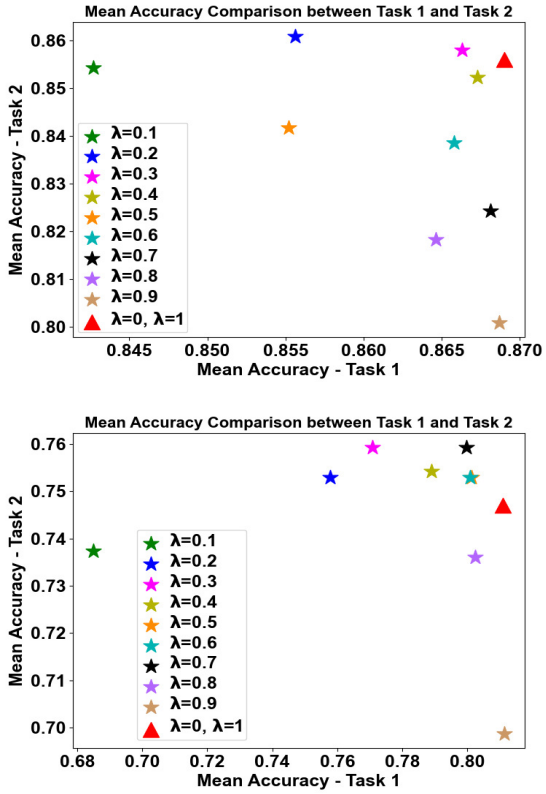

Fig. 2. Average classification accuracy for various λ values on detecting AMP property (Task 1) and alpha-fold (Task 2) for Dataset 1 (above) and Dataset 2 (below). Comparison with single-task accuracy (λ=0, λ=1) is also provided.

λ=0 are given in Table 2. When focusing solely on Task 1 (AMP identification, λ=1), the classifier's performance is comparable to that achieved by the multitask classifier on this specific task for both Dataset 1 and Dataset 2. Conversely, when emphasizing Task 2 (alpha-fold prediction, λ=0), the multitask classifier manages to attain comparable, and in some cases, even slightly improved accuracies for Task 2 across both datasets. These observations highlight the ability of the multitask classifier to operate effectively in scenarios where either task is the primary focus, demonstrating its versatility in addressing diverse application needs. It is crucial to note that, as of our current knowledge, there exists no model that independently classifies both tasks. Consequently, direct comparisons between our multitask classifier and state-of-the-art models are not currently feasible. This highlights the innovative and unique contribution of the multitask classifier in addressing the dual challenges of AMP identification and alpha-fold prediction within a unified framework.

In terms of model capacity, in the single-task classifier scenario, we employ two separate but identical models, each with 34.7K parameters for Dataset 1 and 43.3K parameters for Dataset 2. In contrast, the multitask classifier utilizes a single model with an architecture identical to the single-task models. This single model efficiently manages parameters, maintaining counts nearly identical to their single-task counterparts, with 34.8K parameters for Dataset 1 and 43.5K parameters for Dataset 2, due to an additional linear layer. Furthermore, the multitask classifier not only conserves computational resources but also reduces training and inference time in half. Each single task, when run separately, demands approximately 8 minutes to complete for Dataset 1 and 4 minutes for Dataset 2. In contrast, the multitask classifier requires the same amount of time for both tasks. This time-saving aspect underscores the model's efficiency, as it accomplishes dual tasks within a similar timeframe as for a single task.

6. CONCLUSIONS

This work introduces a novel multitask classifier that relies on k-mers representation and Transformer networks. The proposed classifier reduces complexity by concurrently handling two tasks, capitalizing on the advantages of learning shared representations. It exhibits competitive performance, comparable to single-task classification, enabling an accurate identification of both the AMP property and the alpha-fold structure simultaneously. Experimental evaluation on real protein data highlights the effectiveness of the proposed model by showcasing reduced training times by 50%, without sacrificing significantly individual task performance. This makes our model a promising solution in real-time or resource-constrained environments.

7. REFERENCES

[1] Michael Zasloff, "Antimicrobial peptides of multicellular organisms," *nature*, vol. 415, no. 6870, pp. 389–395, 2002.

[2] Xingjie Pan and Tanja Kortemme, "Recent advances in de novo protein design: Principles, methods, and applications," *Journal of Biological Chemistry*, vol. 296, 2021.

[3] Christian B Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, no. 4096, pp. 223–230, 1973.

[4] Xin Deng, Jesse Eickholt, and Jianlin Cheng, "A comprehensive overview of computational protein disorder prediction methods," *Molecular BioSystems*, vol. 8, no. 1, pp. 114–121, 2012.

[5] Prashant Kumar, Jayachandran N Kizhakkedathu, and Suzana K Straus, "Antimicrobial peptides: diversity, mechanism of action and strategies to improve the activity and biocompatibility in vivo," *Biomolecules*, vol. 8, no. 1, pp. 4, 2018.

[6] Anvita Gupta and James Zou, "Feedback GAN for DNA optimizes protein functions," *Nature Machine Intelligence*, vol. 1, no. 2, pp. 105–111, 2019.

[7] Andrea Giuliani, Giovanna Pirri, and Silvia Nicoletto, "Antimicrobial peptides: an overview of a promising class of therapeutics," *Open Life Sciences*, vol. 2, no. 1, pp. 1–33, 2007.

[8] Pratiti Bhadra, Jielu Yan, Jinyan Li, Simon Fong, and Shirley WI Siu, "Ampep: Sequence-based prediction of antimicrobial peptides using distribution patterns of amino acid properties and random forest," *Scientific reports*, vol. 8, no. 1, pp. 1697, 2018.

[9] Chia-Ru Chung, Ting-Rung Kuo, Li-Ching Wu, Tzong-Yi Lee, and Jorng-Tzong Horng, "Characterization and identification of antimicrobial peptides with different functional activities," *Briefings in bioinformatics*, vol. 21, no. 3, pp. 1098–1114, 2020.

[10] Travis J Lawrence, Dana L Carper, Margaret K Spangler, Alyssa A Carrell, Tomás A Rush, Stephen J Minter, David J Weston, and Jessy L Labbé, "ampeppy 1.0: a portable and accurate antimicrobial peptide prediction tool," *Bioinformatics*, vol. 37, no. 14, pp. 2058–2060, 2021.

[11] Ke Yan, Hongwu Lv, Yichen Guo, Wei Peng, and Bin Liu, "samppred-gat: prediction of antimicrobial peptide by graph attention network and predicted peptide structure," *Bioinformatics*, vol. 39, no. 1, pp. btac715, 2023.

[12] Samuel Karlin and Stephen F Altschul, "Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes.," *Proceedings of the National Academy of Sciences*, vol. 87, no. 6, pp. 2264–2268, 1990.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[14] Benny Chor, David Horn, Nick Goldman, Yaron Levy, and Tim Massingham, "Genomic dna k-mer spectra: models and modalities," *Genome biology*, vol. 10, pp. 1–10, 2009.

[15] Mirko Torrisi, Manaz Kaleel, and Gianluca Pollastri, "Deeper profiles and cascaded recurrent and convolutional neural networks for state-of-the-art protein secondary structure prediction," *Scientific Reports*, vol. 9, Aug. 2019.

[16] Patrick Ngatchou, Anahita Zarei, and A El-Sharkawi, "Pareto multi objective optimization," in *Proceedings of the 13th international conference on, intelligent systems application to power systems*. IEEE, 2005, pp. 84–91.