IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 October 2023, accepted 27 October 2023, date of publication 31 October 2023, date of current version 3 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3328960

RESEARCH ARTICLE

EmbedCaps-DBP: Predicting DNA-Binding Proteins Using Protein Sequence Embedding and Capsule Network

MUHAMMAD KHAERUL NAIM[1,3], TATI RAJAB MENGKO[1], RUKMAN HERTADI[2], AYU PURWARIANTI[1,4], AND MEREDITA SUSANTY[1,5]

[1] School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung 40132, Indonesia
[2] Faculty of Mathematics and Natural Sciences, Bandung Institute of Technology, Bandung 40132, Indonesia
[3] Department of Informatics Engineering, Universal University, Batam 29433, Indonesia
[4] Center for Artificial Intelligence (U-CoE AI-VLB), Bandung Institute of Technology, Bandung 40132, Indonesia
[5] Department of Computer Science, University of Pertamina, Jakarta 12220, Indonesia

Corresponding author: Tati Rajab Mengko (tati.latifa.e.rajab@gmail.com)

This work was supported in part by the 2023 Research, Community Service and Innovation Program, School of Electrical and Information Engineering, Bandung Institute of Technology; and in part by Lembaga Pengelola Dana Pendidikan (LPDP) Republik Indonesia under Grant FR742022288602.

ABSTRACT DNA-binding interactions are an essential biological activity with important functions, such as DNA replication, transcription, repair, and recombination. DNA-binding proteins (DBPs) have been strongly associated with various human diseases, such as asthma, cancer, and HIV/AIDS. Therefore, some DBPs are used in the pharmaceutical industry to produce antibiotics, anticancer drugs, and anti-inflammatory drugs. Most previous methods have used evolutionary information to predict DBPs. However, these methods have high computing costs and produce unsatisfactory results. This study presents EmbedCaps-DBP, a new method for improving DBP prediction. First, we used three protein sequence embeddings (ProtT5, ESM-1b, and ESM-2) to extract learned feature representations from protein sequences. Those embedding methods can capture important information about amino acids, such as biophysics, biochemistry, structure, and domains, that have not been fully utilized in protein annotation tasks. Then, we used a 1D-capsule network (CapsNet) as a classifier. EmbedCaps-DBP significantly outperformed all existing classifiers in training and independent datasets. Based on two independent datasets, EmbedCaps-DBP (ProtT5) achieved 12.65% and 0.33% higher accuracies than a recent predictor on PDB2272 and PDB186, respectively. These results indicate that our proposed method is a promising predictor of DBPs.

INDEX TERMS Capsule network, DNA-binding proteins, deep learning, machine learning, protein sequence embeddings.

I. INTRODUCTION

DNA-binding proteins (DBPs) have recently emerged as a significant research area in protein science due to their central role in many biological processes [1]. DBPs perform various essential intracellular and intercellular functions, including DNA repair and replication; transcriptional regulation; the separation and combination of single-stranded DNA; and other DNA-related biological activities [2]. Several DBPs are critical to understanding human diseases and developing drugs in the pharmaceutical industry. For example, glucocorticoid receptors function as the active components of dexamethasone, which is used to treat allergies, asthma, anti-inflammatory conditions, and autoimmune diseases [3], [4], [5]. In addition, inhibitor DNA binding protein plays a significant role in tumor-related processes, including metastasis, chemoresistance, and angiogenesis [6].

Initially, various experimental methods were developed to identify DBPs, including X-ray crystallography [7], filter binding assays [8], and genomic analyses [9]. However, experimental procedures are expensive and time-consuming [10]. The size of protein databases has nearly doubled

The associate editor coordinating the review of this manuscript and approving it for publication was Yu-Da Lin.

every two years due to the exponential growth caused by next-generation sequencing technologies [11]. Nevertheless, less than one percent of the >220 million proteins in protein databases have experimental functional annotations [12]. Therefore, it is crucial to develop an automatic prediction method for identifying DBPs.

In sequence-based predictions, a protein sequence is typically represented as a multiple sequence alignment to extract the evolutionary information (EI) using a position-specific scoring matrix (PSSM). Many machine-learning-based computational methods that use EI have been developed to predict DBPs, including DNAbinder [13], SVM-PSSM [14], SVM-PSSM-DT [15], DNABINDPROT [16], DR_bind [17], nDNA-prot [18], iDNAPro-PseAAC [19], Local-DPP [20], iDNA-Prot [21], iDNAProt-ES [22], PseDNA-Pro [23], DPP-PseAAC [24], DBPPred-PDSD [25], TargetDBP [26], and Target-DBPPred [10]. However, there are several drawbacks associated with EI. First, compiling the EI has higher computational costs [27]. Second, EI is not available for all proteins, such as dark proteome or intrinsically disordered proteins [28], [29]. Lastly, predictions based on EI may not be able to distinguish between two proteins belonging to the same family because they average over an entire family [30]. In addition, implementing conventional machine learning as a classifier also hinders the performance of these methods when processing large datasets and requires more human intervention in the feature selection procedure [31]. Therefore, many researchers use pre-trained embedding in protein sequence analysis without relying on EI.

Advances in natural language processing (NLP) and the accessibility of supercomputers have led to pre-trained language models being embraced in proteomics, resulting in a radical paradigm shift [32], [33]. The current trend is to train a language model on a large database of unlabeled protein sequences in an unsupervised or semi-supervised manner, enabling the models to learn sequence patterns, functions, and structures. This pre-training provides us with a general understanding of the protein sequence in the form of embeddings that have been found effective in solving various downstream tasks, such as protein function prediction [34], [35], [36], [37], major histocompatibility complex binding prediction [38], [39], [40], [41], protein classification [42], [43], contact map prediction [44], and protein-protein interaction prediction [37], [45], [46]. These downstream tasks adopted various sequence embeddings, including ProtTrans [30], ESM-1b [47], SeqVec [48], and ProtVec [49].

This study introduces a new method named EmbedCaps-DBP. Our study's major contributions are as follows:

1) This study aims to simplify the prediction process by utilizing protein sequence embedding and removing human intervention in feature selection from raw sequences. This approach differs from previous studies that relied on evolutionary information, which faced higher computational costs.
2) We used protein sequence embedding to show the benefit of learned embeddings as feature representations. We explored three embedding models: ProtTrans, ESM-1b, and the most recent ESM-2 [50]. To our knowledge, this study is the first to implement protein sequence embedding for predicting DBPs.
3) This highly accurate approach for DBP prediction has 12.65% and 0.33% greater accuracies than a recent predictor for PDB2272 and PDB186, respectively.
4) Unlike previous studies that only performed well on specific datasets, the proposed method performed exceptionally on large and small datasets.

The rest of this paper is divided into three sections. Section II describes the datasets and applied methods in detail, Section III discusses the experimental results and analysis, and Section IV summarizes the conclusions.

II. MATERIAL

A. DATASETS

Selecting appropriate datasets for training and evaluating a proposed model is a crucial research step. Table 1 lists the two dataset pairs we used: PDB14189-PDB2272 and PDB1075-PDB186. The training dataset (PDB14189) was created by removing 25% of similar sequences and selecting protein sequences 50–6000 amino acids long. The remaining sequences in the training set comprised 7060 non-DBP sequences and 7129 DBP sequences. Initially, the independent dataset (PDB2272) included 1153 DPBs and 1153 non-DBPs. However, the final dataset comprised 1119 non-DBPs and 1153 DBPs after 25% of similar sequences were removed. Similarity between sequences is a measure of their empirical relationship to determine the probability that protein sequences evolved from a common ancestor. The proteins in the training dataset with >40% sequence similarity to proteins in the independent dataset were removed using CD-HIT to prevent homology bias between the training and independent datasets [10], [31], [51], [52], [53].

TABLE 1. Training and independent datasets.

No	Dataset	DBPs	Non-DBPs	Total
1	Training	7129	7060	14189
	Independent	1153	1119	2272
2	Training	525	550	1075
	Independent	93	93	186

PDB1075 was used as the training dataset, and PDB186 as the independent dataset, both having been widely used by existing methods for identifying DBPs [15], [20], [21], [23], [25], [54]. PDB1075 was extracted from the most current Protein Data Bank (PDB) version and constructed by eliminating protein sequences whose length was <50 amino acids [55]. The PISCES 40 software was used to eliminate protein sequences with a similarity of >25%. This training dataset (PDB1075) contained 525 DBPs and 550 non-DBPs. We used the independent dataset (PDB186) as the testing

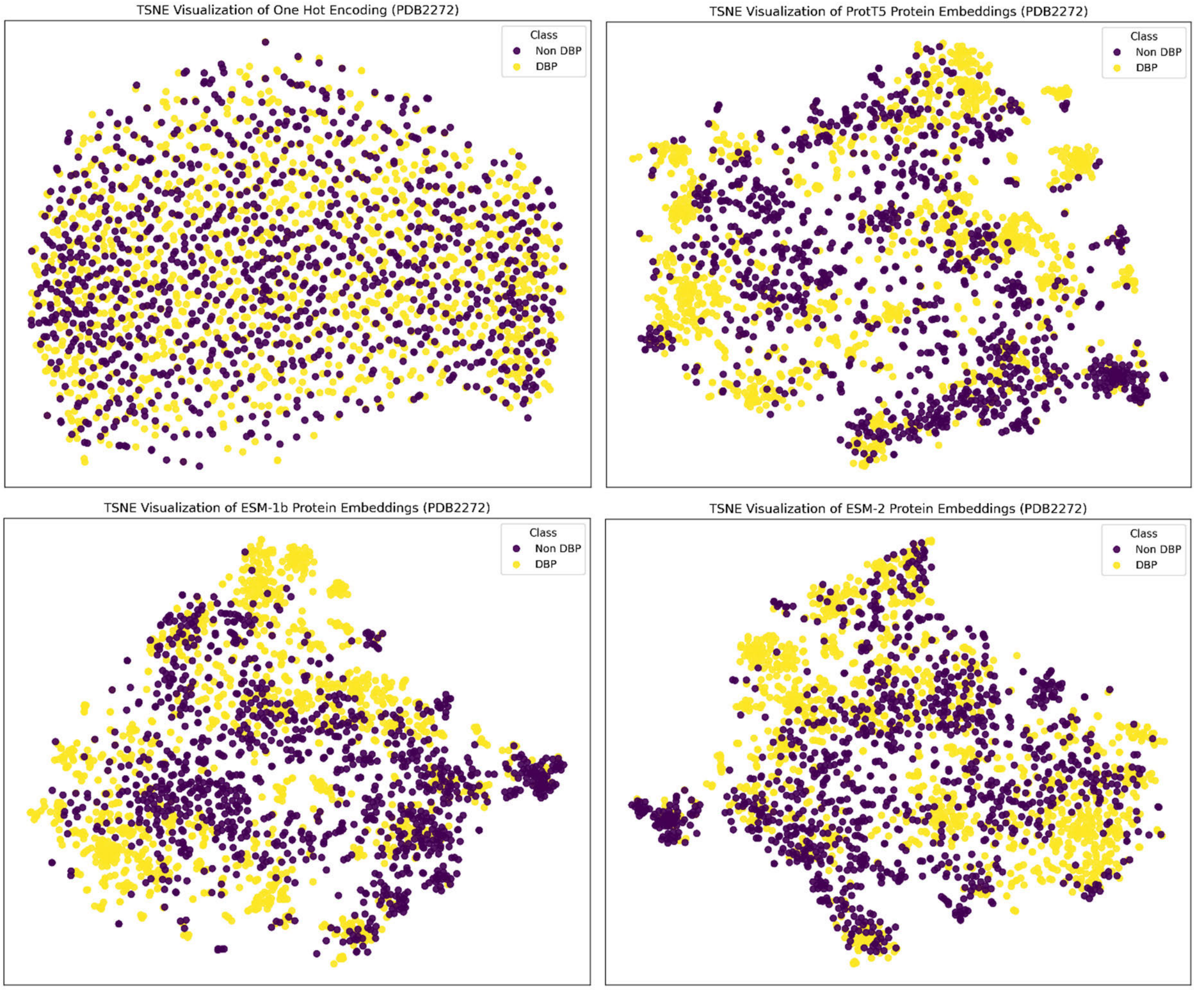

FIGURE 1. Visualization of the DBP embeddings of various models for DBP (yellow) and non-DBP (purple) proteins. All TSNE plots were created with 15000 iterations and a perplexity of 30.

dataset to directly test and compare the performance of our model with other existing methods for prediction. The same stringent criteria were also applied to the construction of this dataset. PDB186 was created by deleting protein sequences whose length was <60 amino acids. NCBI's BLASTCLUST software was used to eliminate protein sequences with a similarity of >25% [56]. The independent dataset (PDB186) contained 93 DBPs and 93 non-DBPs.

B. PROTEIN SEQUENCE EMBEDDINGS

In recent years, NLP and artificial intelligence have shown extraordinary advancements. Large pre-trained language models have radically transformed the NLP field and demonstrated useful capabilities. Previous studies have also demonstrated that protein language models can extract specific functional and structural properties of proteins, such as antibody structure, backbone structure, secondary structure, and tertiary contacts [47], [57], [58], [59].

Since the Transformer's inception has emerged as a versatile tool for language modeling, several Transformer-based models, including ProtTrans [30], ESM [47], and ProteinBERT [60], have proven to be highly competitive compared to other techniques [30], [47], [50], [60]. Most of these models use Bidirectional Encoder Representations from Transformers (BERT)-like architectures and denoising autoencoding training objectives, meaning they are pre-trained by corrupting input tokens and attempting to reconstruct the original sentence [62]. While these models could be modified to generate protein sequences, their most straightforward application is sequence embedding. This study explored three protein sequence embedding models to predict DBPs: ProtTrans, ESM-1b, and ESM-2.

Fig. 1 visualizes the embedding using t-distributed stochastic neighbor embedding (TSNE) plots. TSNE projects n-dimensional embeddings into two dimensions. The visualization of ProtT5, ESM-1b, and ESM-2 in Fig. 1 demonstrates that proteins from the DBP and non-DBP classes with similar properties tend to cluster together. Based on previous studies that have successfully implemented a protein sequence model in downstream tasks, the important information about the protein captured by the protein sequence embedding model, such as biophysical properties, biochemical properties, structure, and domains, is highly useful for the prediction task [35], [37], [38], [44]. In other words, this clustering is learned by pre-trained embeddings, even before seeing the label associated with a particular class. We also demonstrate the visualization of a one-hot encoding for comparison. The TSNE plot for one-hot encoding shows that each protein is separate from the others, making predicting protein classes harder.

C. ProtTrans

ProtTrans offers novel pre-trained models for proteins using various transformer methods. ProtTrans trained four autoencoder models (BERT, Albert, Electra, and T5) and two autoregressive models (Transformer-XL and XLNet) using data from UniRef and BFD containing up to 393 billion amino acids. The ProtTrans model can extract diverse biophysical properties of amino acids, protein structure classes, life and virus domains, and protein functions in conserved motifs. Based on the evaluation in [30], embeddings from their trained ProtTrans (ProtT5) model outperformed state-of-the-art methods in the downstream tasks without using EI for the first time.

This study employs ProtT5 sequence embedding, the best ProtTrans model based on the T5 method [30]. ProtT5 was trained using eight-way model parallelism with approximately three billion learnable parameters on UniRef50. We encoded the protein sequence using the pretrained ProtT5 model. This model accepts the complete protein sequence as input and returns an embedding vector of size 1024 for each protein sequence. The source code for the ProtTrans method is available at: https://github.com/agemagician/ProtTrans/.

D. ESM-1b

The ESM-1b is a self-supervised method for learning protein sequence embeddings by applying transformer models to 250 million protein sequences extracted from the UniParc database, which contains 86 billion amino acids [47]. Initially, 100M-parameter transformer models were trained, and hyperparameter optimization was performed systematically. After determining the optimal set of hyperparameters, the model was scaled to 33 layers with approximately 650 million parameters. In this study, the ESM-1b model generated an embedding vector of size 1280 for each protein sequence. The source code for converting protein sequences to ESM-1b embedding is available at: https://github.com/facebookresearch/esm.

The trained ESM-1b transformer could acquire knowledge of the biochemical properties of amino acids. The generated embeddings allowed the clustering of amino acid residues into several groups consistent with their aromatic, hydrophobic, and polar properties. In addition, charge information and molecular weight were represented throughout the amino acids. The learned embeddings of ESM-1b can also serve as feature representations for various subsequent tasks, such as contact map prediction, protein fitness prediction, protein function prediction, and effects of mutations on protein function [34], [35], [44], [63].

E. ESM-2

The ESM-2 language model is the latest advanced model designed to improve upon the previous model. ESM-2 improves training parameters, model architecture, computational resources, and data compared to ESM-1b. The ESM-2 uses rotary position embedding (RoPE) so that the model can extrapolate beyond the context window on which it was trained. RoPE marginally increases the model's computational cost because it multiplies each query and key vector within the self-attention with a sinusoidal embedding [50].

The ESM-2 was trained with UniRef50 database protein sequences. Fifteen percent of amino acids were masked in an input protein represented as a character sequence of amino acids, and ESM-2 was used to predict the positions of these missing amino acids. Achieving high success requires the model to acquire complex internal representations of inputs, even though this training objective only directly involves amino acid prediction. In ESM-2, these representations are taught to predict secondary structures, contact maps, and binding sites. Like ESM-1b, we generated ESM-2 embedding vectors with a size of 1280 for each protein sequence.

III. METHODS (EMBEDCAPS-DBP)

A. CapsNet

Hinton et al. introduced the capsule to overcome the disadvantages of convolutional neural networks (CNNs) [64]. A CNN's pooling layers lose many important features when extracting and resizing the features. In addition, a CNN cannot learn the relationship between the various extracted features due to the absence of a function capable of acquiring the necessary information [64], [65]. CapsNet uses a squash function, which is similar to the pooling layers in CNNs. This function does not lose any information since it is a nonlinear function that accepts input in vector form and resizes the information in the unit vector without changing its orientation. The mathematical formulas for the capsule's functions are as follows:

$$\hat{u}_{j|i} = W_{ij} u_i \tag{1}$$

where $\hat{u}_{j|i}$ is a prediction vector generated by capsule i that is passed to capsule j and calculated by multiplying the weight matrix W_{ij} (affine transformation matrix) with output u_i of previous capsule layer i.

$$s_j = \sum\nolimits_i c_{ij} \hat{u}_{j|i} \tag{2}$$

where s_j is derived from the sum of the product of $\hat{u}_{j|i}$ and c_{ij}. In CapsNet, conventional CNN neurons are replaced with capsules, and the input and output of each CapsNet unit are converted to vectors. The orientation of the vector represents the properties of a particular entity in the input data. The length of a capsule vector is meant to represent the probability

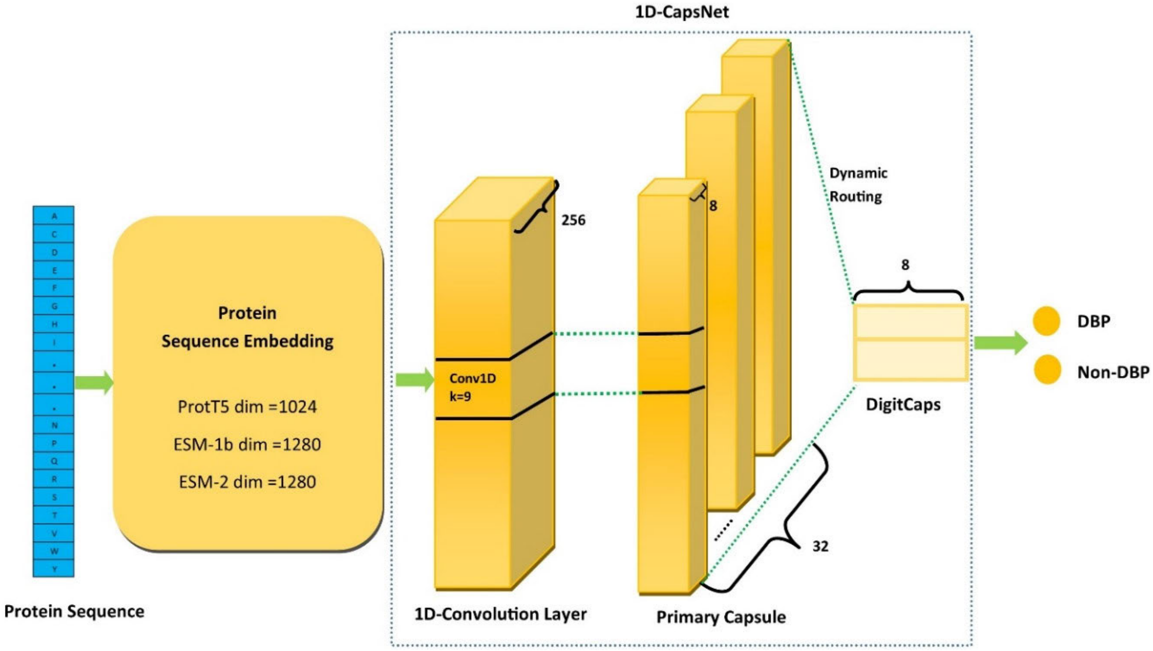

FIGURE 2. The architecture of EmbedCaps-DBP.

that an entity is present in the current input. The "squashing" function, like the CNN activation function, ensures that the vector length is between 0 and 1 (77). The "squashing" function is defined by Eq. (3).

$$v_j = \frac{\|s\|^2}{1 + \|s\|^2} \frac{s_j}{\|s_j\|} \tag{3}$$

where v_j represents the output vector of capsule j and s_j represents the capsule's total input vector.

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_j \exp(b_{ij})} \tag{4}$$

The coupling coefficient c_{ij} in Eq. (2) is determined by the dynamic routing algorithm. Its objective is to allow the input capsule to choose its own path for transmission to the next capsule layer. c_{ij} is determined as the softmax function over b_{ij}, which represents the log prior probability between capsules i and j. CapsNet uses the parameter b_{ij} to determine the relationship between capsules i and j in the previous layers. b_{ij} is initialized to 0 during the initial iteration, and the value of the coupling coefficient c_{ij} is the same for all capsules within a layer. The values of v_j and $\hat{u}_{j|i}$ are then updated using Eqs. (3) and (1), respectively. The parameter b_{ij} is updated by the dot product of $\hat{u}_{j|i}$ and v_j in subsequent iterations [64], [66]:

$$b_{ij} = b_{ij} + \hat{u}_{j|i} v_j \tag{5}$$

When the dot product of $\hat{u}_{j|i}$ and v_j yields a positive result, b_{ij} will have a greater value after being updated using Eq. (5). A greater value for b_{ij} will result in a greater value for c_{ij}, leading to greater values for s_j and v_j, strengthening the connection between capsules i and j. The connection between capsules i and j will be weakened if the dot product of $\hat{u}_{j|i}$ and v_j is negative.

This study designed an EmbedCaps-DPB model for the automatic prediction of DBPs. The proposed model architecture comprises two layers (Fig. 2): an embedding layer and a 1D-CapsNet layer. A protein sequence is first converted to a numerical representation using protein sequence embedding. Three sequence-embedding methods (ProtT5 [dimension 1024), ESM-1b [dimension 1280], and ESM-2 [dimension 1280]) were used to demonstrate the effectiveness of learned embedding as feature representations. We modified the CapsNet by substituting 2D-convolution with 1D-convolution. The 1D-CapsNet layer comprises a convolutional layer, a primary capsule layer, and a DigitCaps layer (Fig. 1). In the convolution layer, the embedding form of the protein sequence is input into a 1D-convolution layer (Conv1D) to extract the feature representation. The primary capsule also implements a 1D-convolution operation with a specific filter size and number executed multiple times to obtain detailed information about the feature. During this process, the squashing function maintains the orientation and length of each capsule between 0 and 1. The number of classifications corresponds to the amount of DigitCaps elements, and the length of each DigitCaps element indicates the probability that the input sequence belongs to the DBP or non-DBP class. Using the dynamic routing method ensures that each element in the primary capsule layers corresponds to a category for a DigitCaps element. In the final step, the category of the input sequence (DBP or non-DBP) is determined by comparing the length of each DigitCaps element.

The proposed method uses several hyperparameters, including batch size, epoch, hidden unit numbers, and routing numbers, that produce the best results. The values of the hyperparameters are summarized in Table 2. Experiments are performed on a public Keras framework using an NVIDIA Tesla T4 GPU with 16 GB of VRAM. The source code of the proposed method is available at: https://github.com/naimji/EmbedCaps-DBP.

TABLE 2. EmbedCaps-DBP hyperparameters.

Hyperparameter	Values
Batch Size	128
Epochs	30
Optimizer	Adam
Routing Number	2
Learning Rate	0.001
Kernel Size	9
Capsule Dimension	8

B. VALIDATION METHODOLOGIES

According to previous research, the following two issues must be considered when evaluating the performance of a new predictor. First, it is necessary to select the metrics that will be used to evaluate the predictor's quality. Second, the appropriate method for calculating the metrics must be determined. We used the six metrics as evaluation metrics: F1-score, accuracy, precision, specificity, sensitivity, the area under the receiver operating characteristic (ROC) curve (AUC), and Matthew's correlation coefficient (MCC). The F1-score was utilized to evaluate dataset imbalance by considering recall and precision value. Accuracy was used because it is the most common measure of classification performance. It is defined as the proportion of correctly classified samples to the total number of samples. Specificity and sensitivity are used because they are commonly used to evaluate classification performance with imbalanced data. MCC was used because it represents the relationship between observed and predicted classifications. AUC was used because it is a significant metric for calculating the prediction model's success rate [1], [10], [18], [21], [67]. These evaluation metrics can be calculated using Eqs. (6)–(11):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{6}$$

$$Precision = \frac{TP}{TP+FP} \tag{7}$$

$$Sensitivity/Recall = \frac{TP}{TP+FN} \tag{8}$$

$$Specificity = \frac{TN}{TN+FP} \tag{9}$$

$$F1-Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

$$MCC = \frac{(TN \times TP)-(FN \times FP)}{\sqrt{(TP+FN)(TP+FP)(TN+FN)(TN+FP)}} \tag{11}$$

where true negative (TN) refers to the number of samples correctly classified as negative, true positive (TP) refers to the number of samples correctly classified as positive, false negative (FN) refers to the number of positive samples misclassified as negative, and false positive (FP) refers to the number of negative samples misclassified as positive. Positive samples are those that contain DBPs, and vice versa. This study used two validation methodologies to evaluate a computational predictor's performance: independent test sets and K-fold cross-validation. We used the five-fold cross-validation method when evaluating the proposed model's performance, which has been shown empirically to produce test error estimates that are neither biased nor highly variable. The final evaluation of the proposed model's performance will be based on the average of the five trials' results.

IV. RESULTS AND DISCUSSION

This section first presents the results of potential classifiers for the proposed method's development. Then, the performance of the proposed method on the benchmark datasets (PDB14189 and PDB 1075) and the independent test datasets (PDB2272 and PDB 186) is reported.

A. RESULTS OF DIFFERENT CLASSIFIERS ON PROTEIN SEQUENCE EMBEDDING

We explored five potential classifiers to determine which is best for predicting DBPs using protein sequence embedding (ProtT5, ESM-1b, and ESM-2): logistic regression (LR), multi-layer perceptron (MLP), one-dimensional CNN (1D-CNN), bidirectional long-short term memory (Bi-LSTM), and 1D-CapsNet. Table 3 shows the individual predictive abilities of each classifier as determined by five-fold cross-validation. Initially, we used simple classifiers such as LR and MLP to demonstrate the benefit of protein sequence embedding as a feature representation for predicting DPBs. We then used a grid search approach to identify the hyperparameters with the highest performance for LR and MLP. The overall performance metrics of LR and MLP outperformed some previous studies that used more complex methods, such as DBP-CNN [1], Local-DPP [20], and MsDBP [31]. We used 1D-CNN, Bi-LSTM, and 1D-CapsNet for the deep learning based-method. The detailed parameters and layers of these deep learning networks are shown in Tables IX-XI in the Appendix. Based on the simulation results, 1D-CapsNet provided the highest F1-Score (94.16%), accuracy (94.71%), sensitivity (96.56%), and MCC (0.88) with the ProtT5 model. In addition, 1D-CapsNet performed better than Bi-LSTM with the ESM-2 model, with a higher F1-Score (by 0.56%), accuracy (by 0.97%), sensitivity (by 2.74%), and MCC (by 0.1). However, its corresponding specificity was 1.92% and lower than Bi-LSTM. Moreover, 1D-CNN performed better than Bi-LSTM with the ProtT5 model, with a higher F1-Score (by 1%), accuracy (by 1%), sensitivity (by 1%), specificity (by 1%), precision (by 1%), and MCC (by 0.03). Table 3 also shows that predictor performance could be further improved by using a deep learning-based method that captures more relevant protein sequence embedding features than LR and MLP.

TABLE 3. Comparison of various existing classifiers on PDB2272.

Embedding	Method	F1-Score (%)	Accuracy (%)	Sensitivity (%)	Specificity (%)	MCC
ProtT5	LR	84.48(±0.78)	83.25(±0.77)	89.68(±0.94)	76.61(±0.94)	0.67(±1.52)
	MLP	86.97(±0.69)	85.32(±0.73)	96.36(±0.57)	73.92(±0.57)	0.72(±1.29)
	1D-CNN	91.88(±1.78)	91.88(±1.58)	91.88(±1.77)	**91.86(±1.78)**	0.84(±3.55)
	Bi-LSTM	90.88(±2.26)	90.88(±2.01)	90.88(±2.34)	90.86(±2.26)	0.81(±4.51)
	1D-CapsNet	**94.16(±0.18)**	**94.71(±0.13)**	**96.56(±0.04)**	91.43(±0.42)	**0.88(±0.35)**
ESM-1b	LR	85.59(±0.76)	84.53(±0.76)	90.45(±0.87)	78.41(±0.87)	0.69(±1.49)
	MLP	85.56(±0.70)	84.42(±0.71)	90.72(±0.62)	77.96(±0.62)	0.69(±1.30)
	1D-CNN	87.64(±0.45)	87.64(±0.41)	87.64(±0.46)	87.59(±0.45)	0.75(±0.91)
	Bi-LSTM	**93.22(±0.81)**	93.21(±0.71)	93.22(±0.79)	**93.21(±0.81)**	**0.86(±1.61)**
	1D-CapsNet	92.77(±0.43)	**93.26 (±0.39)**	**94.77(±0.46)**	90.44(±0.52)	0.85(±0.88)
ESM-2	LR	86.02(±0.77)	85.23(±0.76)	89.41(±0.92)	80.91(±0.92)	0.71(±1.51)
	MLP	88.38(±0.71)	87.35(±0.71)	94.71(±0.85)	79.75(±0.85)	0.75(±1.39)
	1D-CNN	91.17(±0.93)	91.17(±0.83)	91.18(±0.92)	91.16(±0.93)	0.82(±1.86)
	Bi-LSTM	93.14(±0.53)	93.14 (±0.47)	93.14(±0.53)	**93.13(±0.53)**	0.86(±1.05)
	1D-CapsNet	**93.70(±0.28)**	**94.11 (± 0.26)**	**95.88(±0.43)**	91.21(±0.33)	**0.87(±0.58)**

Key: Cross validation result 'mean (±standard deviation)'; MCC, Matthew's correlation coefficient.

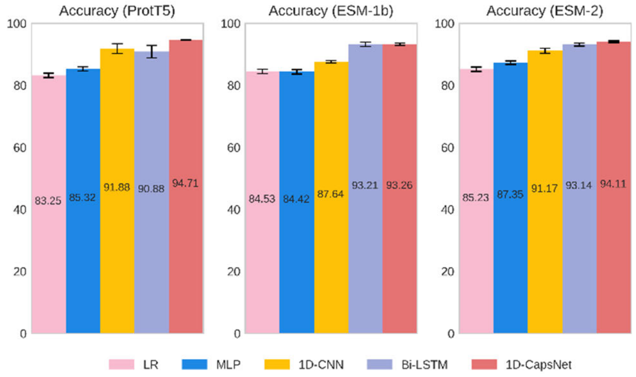

FIGURE 3. The accuracy of different classifiers.

TABLE 4. Comparison with existing methods on a training set (PDB14189).

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	MCC
DNA-Prot	72.55	82.67	59.76	0.44
iDNA-Prot	75.40	83.81	64.73	0.50
iDNA-Prot|dis	77.30	79.40	75.27	0.54
MsDBP	80.29	80.87	79.72	0.60
DBP-CNN	83.09	80.66	85.54	0.66
Target-DBPPred	86.96	86.59	86.82	0.71
EmbedCaps-DBP (ProtT5)	99.21	**99.87**	98.65	**0.99**
EmbedCaps-DBP (ESM-1b)	99.09	99.74	98.42	0.98
EmbedCaps-DBP (ESM-2)	**99.24**	99.85	**98.69**	**0.99**

Key: MCC, Matthew's correlation coefficient

To demonstrate the robustness of 1D-CapsNet, we also compared the standard deviation of each classifier. According to Table 3, 1D-CapsNet has the lowest standard deviation among all metric evaluations for all protein sequence embedding models. For example, based on Fig. 3, 1D-CapsNet achieved the best overall accuracy with the lowest standard deviation (0.13 with ProtT5, 0.36 with ESM-1b, and 0.26 with ESM-2) compared to all other classifiers. On the ESM-1b model, Bi-LSTM has slightly better performance than 1D-CapsNet, with a higher F1 score (by 0.45%), specificity (by 2.77%), and MCC (by 0.01) (Table 3). Nevertheless, Bi-LSTM had a higher standard deviation than 1D-CapsNet. In addition, 1D-CapsNet achieved this result with approximately one-eighth the number of parameters required by Bi-LSTM. These results indicate that the ability of 1D-CapsNet to capture the relationship between features has a greater potential to produce a more accurate representation and understanding of a given annotation task than the other classifiers.

B. PERFORMANCE COMPARISON WITH EXISTING PREDICTORS USING PDB14189 AND PDB2272

The EmbedCaps-DBP method's performance was compared with state-of-the-art methods using the same training and independent test datasets to ensure a fair evaluation. Table 4 shows that we compared our proposed method with DNA-Prot [67], iDNA-Prot [21], iDNA-Prot|dis [68], MsDBP [31], DBP-CNN [1], and Target-DBPPred [10]. Most of these methods are sequence-based, with EI as their input. The EmbedCaps-DBP with ProtT5, ESM-1b, and ESM-2 models outperformed all existing state-of-the-art methods with all evaluation metrics. The ESM-2 model achieved the best performance among protein sequence embedding models with 12.28%, 13.26%, 11.87%, and 0.28 improvements in accuracy, sensitivity, specificity, and MCC, respectively, over Target-DBPPred. Target-DBPPred is currently the highest-performing among the existing classifiers that employ a multi-evolutionary approach to extract diverse EI features from protein sequences. Similarly, the ProtT5 and ESM-1b models improved accuracy, sensitivity, specificity, and MCC by >10%.

The proposed method's quality was assessed based on its optimal generalizability to unobserved datasets. This study used an independent dataset (PDB2272) to validate the proposed method. Table 5 compares the accuracy of its predictions with eight previous methods: DNA-Prot [67], iDNA-Prot [21], iDNA-Prot|dis [68], Local-DPP [20], MsDBP [31],

TABLE 5. Comparison with existing methods on an independent test set (PDB2272).

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	AUC (%)	MCC
DNA-Prot	61.80	69.90	53.80	-	0.24
iDNA-Prot	67.20	67.70	66.70	-	0.34
iDNA-Prot|dis	76.90	79.60	74.20	-	0.53
Local-DPP	50.57	8.76	**93.66**	-	0.04
MsDBP	66.99	70.69	63.18	-	0.33
DBP-CNN	67.91	69.04	66.76	-	0.35
HKAM-MKM	78.43	94.02	62.38	-	0.59
Target-DBPPred	82.06	87.10	76.78	-	0.63
EmbedCaps-DBP (ProtT5)	**94.71**	**96.56**	91.43	**97.76**	**0.88**
EmbedCaps-DBP (ESM-1b)	93.26	94.77	90.44	96.93	0.85
EmbedCaps-DBP (ESM-2)	94.11	95.88	91.21	97.41	0.87

Key: AUC, the area under the receiver operating characteristic curve; MCC, Matthew's correlation coefficient

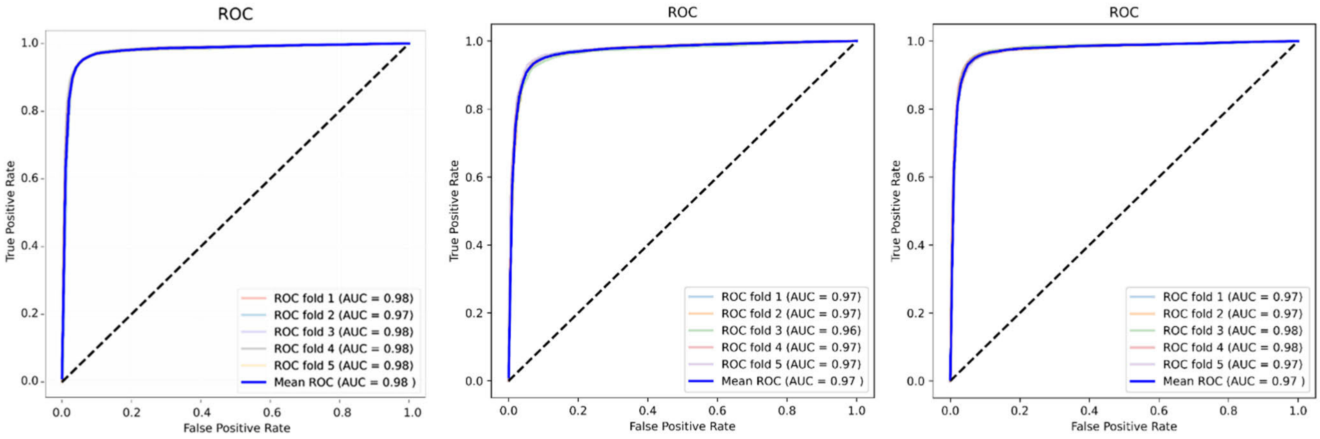

FIGURE 4. ROC curves for EmbedCaps-DBP (PDB2272) with (a) ProtT5, (b) ESM-1b, and (c) ESM-2.

TABLE 6. Comparison with existing methods on a training set (PDB1075).

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	MCC
PSSM-DT	79.96	81.91	78.00	0.62
DNAbinder	73.58	66.47	80.36	0.47
PseDNA-Pro	76.55	79.61	73.63	0.53
Local-DPP	79.20	84.00	74.50	0.59
FKRR-MVSF	83.26	85.71	80.91	0.67
HMMBinder	86.33	87.07	85.55	0.72
DBPPred-PDSD	89.02	89.14	88.88	0.78
HKAM-MKM	84.28	80.00	88.76	0.69
EmbedCaps-DBP (ProtT5)	99.08	99.94	97.75	0.97
EmbedCaps-DBP (ESM-1b)	**99.54**	**100**	**99.27**	**0.99**
EmbedCaps-DBP (ESM-2)	99.04	99.94	98.29	0.98

DBP-CNN [51], Target-DBPPred [10], and HKAM-MKM [69]. EmbedCaps-DBP (ProtT5) performed better than all existing classifiers and had higher accuracy (by 12.65%), sensitivity (by 9.46%), specificity (by 14.65%), and MCC (by 0.25) than Target-DBPPred. However, its corresponding specificity values were 2.23% lower than Local-DPP.

Two deep learning-based classifiers, MsDBP and DBP-CNN, performed poorly on PDB2272, demonstrating the difficulty of this task. In contrast, our proposed EmbedCaps-DBP method achieved very significant improvements in the training and independent test sets with all protein sequence embedding models, making it a promising predictor.

Figs. 4a, 4b, and 4c show the ROC curves for EmbedCaps-DBP with the ProtT5, ESM-1b, and ESM-2 models, respectively, while evaluating the predictions on the PDB2272 test set. All ROC curves in Fig. 3 demonstrate the capability of EmbedCaps-DBP to achieve a high TP rate of $\geq$ 93% (rate of correct DBP predictions) with a very low FP rate of around 10%. EmbedCaps-DBP with the ProtT5 model had the highest average AUC of 97.76%, 0.83% higher than with the ESM-1b model and 0.35% higher than with the ESM-2 model. The small performance gap between the three embedding models reflects their almost identical data distributions in the TSNE visualization in Fig. 1.

C. PERFORMANCE COMPARISON WITH EXISTING PREDICTORS USING PDB1075 AND PDB186

The generalizability of EmbedCaps-DBP was further evaluated using PDB1075 as the training set and PDB186 as the independent test set. The small number of samples is one of the challenges of using the PDB1075 and PDB186

TABLE 7. Comparison with existing methods on an independent test set (PDB186).

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	AUC (%)	MCC
PSSM-DT	80.00	87.09	72.83	-	0.65
DNAbinder	60.80	57.00	64.50	-	0.22
PseDNA-Pro	72.00	79.50	64.50	-	0.45
Local-DPP	79.00	92.50	65.60	-	0.62
FKRR-MVSF	81.70	98.90	64.50	-	0.68
HMMBinder	69.02	61.53	76.34		0.39
DBPPred-PDSD	81.72	95.69	67.39	-	0.65
HKAM-MKM	87.10	**100**	74.19		**0.77**
EmbedCaps-DBP (ProtT5)	**87.43**	91.11	**77.90**	91.79	0.70
EmbedCaps-DBP (ESM-1b)	84.78	91.03	76.88	**92.16**	0.69
EmbedCaps-DBP (ESM-2)	85.67	90.71	77.72	90.92	0.69

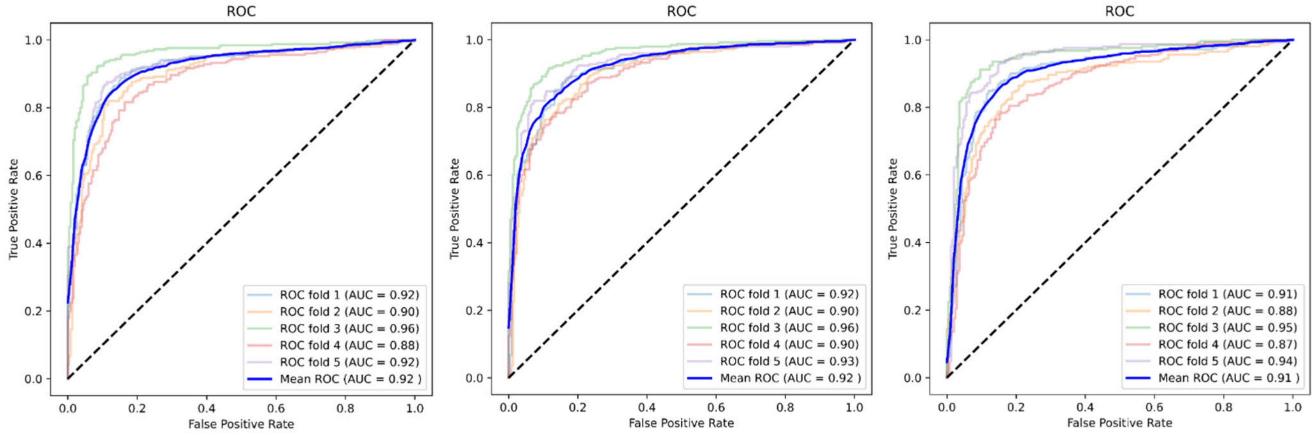

FIGURE 5. ROC curves for EmbedCaps-DBP (PDB186) with (a) ProtT5, (b) ESM-1b, and (c) ESM-2.

TABLE 8. Comparison with deep learning methods on PDB186.

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	MCC
1D-CNN (One-hot)	51.63	51.63	51.68	0.34
1D-CNN (word2vec)	57.32	57.32	57.47	0.15
1D-CNN (ProtT5)	75.98	75.98	76.19	0.52
1D-CNN (ESM-1b)	61.16	61.16	61.57	0.23
1D-CNN (ESM-2)	65.59	65.59	65.69	0.31
Bi-LSTM (One-hot)	68.45	68.45	68.77	0.38
Bi-LSTM (word2vec)	60.59	60.59	60.79	0.22
Bi-LSTM (ProtT5)	78.78	78.78	78.87	0.58
Bi-LSTM (ESM-1b)	78.45	78.45	78.53	0.57
Bi-LSTM (ESM-2)	80.63	80.63	**80.32**	0.61
1D-CapsNet (One-hot)	71.66	86.55	51.73	0.41
1D-CapsNet (word2vec)	68.21	82.55	48.37	0.33
EmbedCaps-DBP (ProtT5)	**87.43**	**91.11**	77.90	**0.70**
EmbedCaps-DBP (ESM-1b)	84.78	91.03	76.88	0.69
EmbedCaps-DBP (ESM-2)	85.67	90.71	77.72	0.69

datasets, particularly with deep learning methods that require large datasets for optimal performance. Consequently, most classifiers developed for this dataset are based on machine learning, which requires more human intervention in the feature selection procedure. We compared the proposed method with eight existing classifiers: PSSM-DT [15], DNAbinder [13], Local-DPP [20], PseDNA-Pro [23], FKRR-MVSF [70], HMMBinder [54], DBPPred-PDSD [25], and HKAM-MKM [69]. Table 6 shows that EmbedCaps-DBP outperformed all existing methods with remarkable results with all protein sequence embedding models in the training set (PDB1075). EmbedCaps-DBP (ESM-1b) performed best among the protein sequence models and provided 10.52%, 10.86%, 10.39%, and 0.21 improvements in accuracy, sensitivity, specificity, and MCC, respectively, over DBPPred-PDSD.

Table 7 shows the performance result for EmbedCaps-DBP using PDB186 as the independent dataset. EmbedCaps-DBP (ProtT5) achieved the highest accuracy (87.43%) and specificity (77.90%) compared to the recent predictor HKAM-MKM. The machine learning-based HKAM-MKM method uses six different feature extraction algorithms. Four algorithms are used to capture EI, and two are used to extract physicochemical properties. HKAM-MKM had the highest sensitivity (100%) and MCC (0.77). However, it has a higher computational cost than our proposed method. Figs. 5a, 5b, and 5c show that ROC curves and AUC values vary among folds with PDB186 due to the small number of data samples. The proposed method achieved a TP rate of >91% with an FP rate of approximately 23%. EmbedCaps-DBP with the ESM1b model had an average AUC of 92.16%, 0.37% higher

TABLE 9. Detail of Layers and Parameters of 1D-CapsNet.

Layer	Hyperparameters	Output shape	Parameter no.
Input Layer	-	(None, 1024,1)	0
Convolution (Conv1D)	Filter =256, Kernel =9	(None, 1016, 256)	2560
Primary Capsule	Capsule Dim =8	(None, 504, 256)	590080
Primary Capsule (Reshape)	-	(None, 16128, 8)	0
Primary Capsule (Squashing function)	Number of routings =2	(None, 16128,8)	0
DigitCaps	-	(None, 2, 5)	2064384
Output	-	(None, 2)	0
Total			**2657024**

TABLE 10. Detail of Layers and Parameters of Bi-LSTM.

Layer	Hyperparameters	Output shape	Parameter no.
Input Layer	-	(None, 1024,1)	0
Bidirectional	Hidden Units= 64	(None, 1024, 128)	33792
Flatten	-	(None, 131072)	256
Dense	-	(None, 128)	16777344
Batch Normalization	-	(None, 128)	512
Dropout	Dropout=0.3	(None, 128)	0
Dense	-	(None, 2)	258
Total	-		**16811906**

TABLE 11. Detail of Layers and Parameters of 1D-CNN.

Layer	Hyperparameters	Output shape	Parameter
Input Layer		(None, 1024,1)	0
Convolution_Layer_1 (Conv1D)	Filter =64, Kernel =3, padding= same, activation= relu	(None, 1024, 64)	256
Batch Normalization	-	(None, 1024, 64)	256
Max Pooling	Pooling size =5	(None, 204, 64)	0
Dropout	Dropout=0.5	(None, 204,64)	0
Convolution_Layer_2 (Conv1D)	Filter =64, Kernel =3, padding= same, activation= relu	(None, 204, 64)	12352
Batch Normalization	-	(None, 204, 64)	256
Max Pooling	Pooling size =5	(None, 40, 64)	0
Dropout	Dropout=0.5	(None, 40, 64)	0
Flatten	-	(None, 2560)	0
Dense	-	(None, 128)	327808
Batch Normalization	-	(None, 128)	512
Dropout	Dropout=0.3	(None, 128)	0
Dense	-	(None, 2)	258
Total			**341698**

than with the ProtT5 model and 1.24% higher than with the ESM-2 model.

D. DISCUSSION

A method that can be implemented in a real-world scenario must be applicable to various datasets. However, most previous methods only perform well on small or large datasets. For example, Local-DPP and HKAM-MKM perform better with small datasets than large ones due to the limitations of conventional machine learning-based classifiers. Target-DBPPred is a machine learning-based method designed for large datasets that performed better than deep learning-based classifiers MsDBP and DBP-CNN. However, its performance remained 12.65% below our proposed method. This considerable performance gap demonstrates the weakness of machine learning methods on large datasets.

The proposed method performed exceptionally well on both datasets. CapsNet is a deep learning method that performed better on small datasets than CNN and Bi-LSTM due to its ability to capture the relationship between features [71], [72], [73]. To confirm this, we compared CapsNet with 1D-CNN and Bi-LSTM on PDB186 (Table 8).

The results show that our method achieved higher accuracy, sensitivity, and MCC than Bi-LSTM. In addition, we compared the protein sequence embedding model with one-hot encoding and word2vec to demonstrate how effectively it captures essential information about proteins. Based on table 8, simulation results indicate that protein sequence embedding outperforms one hot encoding and word2vec by an average of more than 10% for all evaluation metrics.

The ability of protein sequence embedding to provide important information has proven advantageous for predicting DBPs than EI. By combining protein sequence embedding and CapsNet, we designed a new method that was more successful than other existing classifiers in distinguishing DBPs from non-DBPs due to its robustness and generalizability across all tested datasets.

V. CONCLUSION

This study introduced EmbedCaps-DBP, a novel computational method for improving DBP prediction performance. EmbedCaps-DBP uses protein sequence embeddings as feature representations and 1D-CapsNet methods to capture the relationship between features to predict DBPs. Based on our experiment results, EmbedCaps-DBP significantly outperformed all existing classifiers in both dataset pairs. The EmbedCaps-DBP (ProtT5) method achieved an accuracy of 94.71% with PDB2272 and 87.43% with PDB186. The proposed method could potentially be a valuable tool for the proteomic analysis of DNA binding sites, particularly in humans. This research will assist medical professionals in developing advanced and early diagnostic methods for diseases such as allergies, asthma, HIV/AIDS, and cancers. It will also benefit the pharmaceutical industry in producing anticancer drugs, antibiotics, steroids, and anti-inflammatory drugs at a low cost. Future research can focus on combining protein sequence embedding models with more complex classifiers to enhance prediction performance in small datasets.

APPENDIX A

See Tables 9–11.

REFERENCES

[1] O. Barukab, F. Ali, W. Alghamdi, Y. Bassam, and S. A. Khan, "DBP-CNN: Deep learning-based prediction of DNA-binding proteins by coupling discrete cosine transform with two-dimensional convolutional neural network," *Expert Syst. Appl.*, vol. 197, Jul. 2022, Art. no. 116729, doi: 10.1016/J.ESWA.2022.116729.

[2] S. Gattani, A. Mishra, and M. T. Hoque, "StackCBPred: A stacking based prediction of protein-carbohydrate binding sites from sequence," *Carbohydrate Res.*, vol. 486, Dec. 2019, Art. no. 107857, doi: 10.1016/j.carres.2019.107857.

[3] J. P. Overington, B. Al-Lazikani, and A. L. Hopkins, "How many drug targets are there?" *Nature Rev. Drug Discovery*, vol. 5, no. 12, pp. 993–996, Dec. 2006, doi: 10.1038/nrd2199.

[4] H. Gronemeyer, J.-Å. Gustafsson, and V. Laudet, "Principles for modulation of the nuclear receptor superfamily," *Nature Rev. Drug Discovery*, vol. 3, no. 11, pp. 950–964, Nov. 2004, doi: 10.1038/nrd1551.

[5] W. H. Hudson, I. M. S. D. Vera, J. C. Nwachukwu, E. R. Weikum, A. G. Herbst, Q. Yang, D. L. Bain, K. W. Nettles, D. J. Kojetin, and E. A. Ortlund, "Cryptic glucocorticoid receptor-binding sites pervade genomic NF-κB response elements," *Nature Commun.*, vol. 9, no. 1, p. 1337, Apr. 2018, doi: 10.1038/s41467-018-03780-1.

[6] P. A. Hoskisson and S. Rigali, "Variation in form and function," in *Advances in Applied Microbiology*. USA: Academic, 2009, ch. 1, pp. 1–22, doi: 10.1016/S0065-2164(09)69001-8.

[7] R. Jaiswal, S. K. Singh, D. Bastia, and C. R. Escalante, "Crystallization and preliminary X-ray characterization of the eukaryotic replication terminator Reb1-ter DNA complex," *Acta Crystallographica F Struct. Biol. Commun.*, vol. 71, no. 4, pp. 414–418, Apr. 2015, doi: 10.1107/S2053230X15004112.

[8] G. Nimrod, A. Szilágyi, C. Leslie, and N. Ben-Tal, "Identification of DNA-binding proteins using structural, electrostatic and evolutionary features," *J. Mol. Biol.*, vol. 387, no. 4, pp. 1040–1053, Apr. 2009, doi: 10.1016/j.jmb.2009.02.023.

[9] K. Freeman, M. Gwadz, and D. Shore, "Molecular and genetic analysis of the toxic effect of RAP1 overexpression in yeast," *Genetics*, vol. 141, no. 4, pp. 1253–1262, Dec. 1995, doi: 10.1093/genetics/141.4.1253.

[10] F. Ali, H. Kumar, S. Patil, K. Kotecha, A. Banjar, and A. Daud, "Target-DBPPred: An intelligent model for prediction of DNA-binding proteins using discrete wavelet transform based compression and light eXtreme gradient boosting," *Comput. Biol. Med.*, vol. 145, Jun. 2022, Art. no. 105533, doi: 10.1016/j.compbiomed.2022.105533.

[11] The UniProt Consortium, "UniProt: A worldwide hub of protein knowledge," *Nucleic Acids Res*, vol. 47, no. D1, pp. D506–D515, 2018, doi: 10.1093/nar/gky1049.

[12] Y. Jiang et al., "An expanded evaluation of protein function prediction methods shows an improvement in accuracy," *Genome Biol.*, vol. 17, no. 1, p. 184, Dec. 2016, doi: 10.1186/s13059-016-1037-6.

[13] M. Kumar, M. M. Gromiha, and G. P. Raghava, "Identification of DNA-binding proteins using support vector machines and evolutionary profiles," *BMC Bioinf.*, vol. 8, no. 1, p. 463, Dec. 2007, doi: 10.1186/1471-2105-8-463.

[14] S.-Y. Ho, F.-C. Yu, C.-Y. Chang, and H.-L. Huang, "Design of accurate predictors for DNA-binding sites in proteins using hybrid SVM-PSSM method," *Biosystems*, vol. 90, no. 1, pp. 234–241, Jul. 2007, doi: 10.1016/j.biosystems.2006.08.007.

[15] R. Xu, J. Zhou, H. Wang, Y. He, X. Wang, and B. Liu, "Identifying DNA-binding proteins by combining support vector machine and PSSM distance transformation," *BMC Syst. Biol.*, vol. 9, no. S1, p. S10, Dec. 2015, doi: 10.1186/1752-0509-9-S1-S10.

[16] P. Ozbek, S. Soner, B. Erman, and T. Haliloglu, "DNABINDPROT: Fluctuation-based predictor of DNA-binding residues within a network of interacting residues," *Nucleic Acids Res.*, vol. 38, no. 2, pp. W417–W423, Jul. 2010, doi: 10.1093/nar/gkq396.

[17] Y. C. Chen, J. D. Wright, and C. Lim, "DR_bind: A web server for predicting DNA-binding residues from the protein structure based on electrostatics, evolution and geometry," *Nucleic Acids Res.*, vol. 40, no. W1, pp. W249–W256, Jul. 2012, doi: 10.1093/nar/gks481.

[18] L. Song, D. Li, X. Zeng, Y. Wu, L. Guo, and Q. Zou, "NDNA-prot: Identification of DNA-binding proteins based on unbalanced classification," *BMC Bioinf.*, vol. 15, no. 1, p. 298, Dec. 2014, doi: 10.1186/1471-2105-15-298.

[19] B. Liu, S. Wang, and X. Wang, "DNA binding protein identification by combining pseudo amino acid composition and profile-based protein representation," *Sci. Rep.*, vol. 5, no. 1, p. 15479, Oct. 2015, doi: 10.1038/srep15479.

[20] L. Wei, J. Tang, and Q. Zou, "Local-DPP: An improved DNA-binding protein prediction method by exploring local evolutionary information," *Inf. Sci.*, vol. 384, pp. 135–144, Apr. 2017, doi: 10.1016/j.ins.2016.06.026.

[21] W.-Z. Lin, J.-A. Fang, X. Xiao, and K.-C. Chou, "IDNA-prot: Identification of DNA binding proteins using random forest with grey model," *PLoS ONE*, vol. 6, no. 9, Sep. 2011, Art. no. e24756, doi: 10.1371/journal.pone.0024756.

[22] S. Y. Chowdhury, S. Shatabda, and A. Dehzangi, "IDNAProt-ES: Identification of DNA-binding proteins using evolutionary and structural features," *Sci. Rep.*, vol. 7, no. 1, p. 14938, Nov. 2017, doi: 10.1038/s41598-017-14945-1.

[23] B. Liu, J. Xu, S. Fan, R. Xu, J. Zhou, and X. Wang, ''PseDNA-pro: DNA-binding protein identification by combining Chou's PseAAC and physicochemical distance transformation,'' *Mol. Informat.*, vol. 34, no. 1, pp. 8–17, Jan. 2015, doi: 10.1002/minf.201400025.

[24] M. S. Rahman, S. Shatabda, S. Saha, M. Kaykobad, and M. S. Rahman, ''DPP-PseAAC: A DNA-binding protein prediction model using Chou's general PseAAC,'' *J. Theor. Biol.*, vol. 452, pp. 22–34, Sep. 2018, doi: 10.1016/j.jtbi.2018.05.006.

[25] F. Ali, M. Kabir, M. Arif, Z. N. Khan Swati, Z. U. Khan, M. Ullah, and D.-J. Yu, ''DBPPred-PDSD: Machine learning approach for prediction of DNA-binding proteins using discrete wavelet transform and optimized integrated features space,'' *Chemometric Intell. Lab. Syst.*, vol. 182, pp. 21–30, Nov. 2018, doi: 10.1016/j.chemolab.2018.08.013.

[26] J. Hu, X. Zhou, Y.-H. Zhu, D.-J. Yu, and G. Zhang, ''TargetDBP: Accurate DNA-binding protein prediction via sequence-based multi-view feature learning,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 17, no. 4, pp. 1419–1429, Jul./Aug. 2019, doi: 10.1109/TCBB.2019.2893634.

[27] M. Bernhofer et al., ''PredictProtein-predicting protein structure and function for 29 years,'' *Nucleic Acids Res.*, vol. 49, no. W1, pp. W535–W540, Jul. 2021, doi: 10.1093/nar/gkab354.

[28] N. Perdigão, J. Heinrich, C. Stolte, K. S. Sabir, M. J. Buckley, B. Tabor, B. Signal, B. S. Gloss, C. J. Hammang, B. Rost, A. Schafferhans, and S. I. O'Donoghue, ''Unexpected features of the dark proteome,'' *Proc. Nat. Acad. Sci. USA*, vol. 112, no. 52, pp. 15898–15903, Dec. 2015, doi: 10.1073/pnas.1508380112.

[29] P. Radivojac, Z. Obradovic, D. K. Smith, G. Zhu, S. Vucetic, C. J. Brown, J. D. Lawson, and A. K. Dunker, ''Protein flexibility and intrinsic disorder,'' *Protein Sci.*, vol. 13, no. 1, pp. 71–80, Jan. 2004, doi: 10.1110/ps.03128904.

[30] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, D. Bhowmik, and B. Rost, ''ProtTrans: Toward understanding the language of life through self-supervised learning,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 7112–7127, Oct. 2022, doi: 10.1109/TPAMI.2021.3095381.

[31] X. Du, Y. Diao, H. Liu, and S. Li, ''MsDBP: Exploring DNA-binding proteins by integrating multiscale sequence information via Chou's five-step rule,'' *J. Proteome Res.*, vol. 18, no. 8, pp. 3119–3132, Aug. 2019, doi: 10.1021/acs.jproteome.9b00226.

[32] D. W. Otter, J. R. Medina, and J. K. Kalita, ''A survey of the usages of deep learning for natural language processing,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 604–624, Feb. 2021, doi: 10.1109/TNNLS.2020.2979670.

[33] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, ''Pre-trained models for natural language processing: A survey,'' *Sci. China Technol. Sci.*, vol. 63, no. 10, pp. 1872–1897, Oct. 2020, doi: 10.1007/s11431-020-1647-3.

[34] C. Hsu, H. M. Nisonoff, C. Fannjiang, and J. Listgarten, ''Combining evolutionary and assay-labelled data for protein fitness prediction,'' *Nature Biotechnol.*, vol. 40, no. 7, pp. 1114–1122, 2022, doi: 10.1038/s41587-021-01146-5.

[35] B. Lai and J. Xu, ''Accurate protein function prediction via graph attention networks with predicted structure information,'' *Briefings Bioinf.*, vol. 23, no. 1, Jan. 2022, Art. no. bbab502, doi: 10.1093/bib/bbab502.

[36] A. Villegas-Morcillo, S. Makrodimitris, R. C. H. J. van Ham, A. M. Gomez, V. Sanchez, and M. J. T. Reinders, ''Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function,'' *Bioinformatics*, vol. 37, no. 2, pp. 162–170, Apr. 2021, doi: 10.1093/bioinformatics/btaa701.

[37] M. Littmann, M. Heinzinger, C. Dallago, T. Olenyi, and B. Rost, ''Embeddings from deep learning transfer GO annotations beyond homology,'' *Sci. Rep.*, vol. 11, no. 1, p. 1160, Jan. 2021, doi: 10.1038/s41598-020-80786-0.

[38] H. Zeng and D. K. Gifford, ''DeepLigand: Accurate prediction of MHC class i ligands using peptide embedding,'' *Bioinformatics*, vol. 35, no. 14, pp. i278–i283, Jul. 2019, doi: 10.1093/bioinformatics/btz330.

[39] J. Cheng, K. Bendjama, K. Rittner, and B. Malone, ''BERTMHC: Improved MHC-peptide class II interaction prediction with transformer and multiple instance learning,'' *Bioinformatics*, vol. 37, no. 22, pp. 4172–4179, Nov. 2021, doi: 10.1093/bioinformatics/btab422.

[40] J. Vielhaben, M. Wenzel, W. Samek, and N. Strodthoff, ''USMPep: Universal sequence models for major histocompatibility complex binding affinity prediction,'' *BMC Bioinf.*, vol. 21, no. 1, p. 279, Dec. 2020, doi: 10.1186/s12859-020-03631-1.

[41] P. Phloyphisut, N. Pornputtapong, S. Sriswasdi, and E. Chuangsuwanich, ''MHCSeqNet: A deep neural network model for universal MHC binding prediction,'' *BMC Bioinf.*, vol. 20, no. 1, p. 270, Dec. 2019, doi: 10.1186/s12859-019-2892-4.

[42] S. M. A. Islam, B. J. Heil, C. M. Kearney, and E. J. Baker, ''Protein classification using modified n-grams and skip-grams,'' *Bioinformatics*, vol. 34, no. 9, pp. 1481–1487, May 2018, doi: 10.1093/bioinformatics/btx823.

[43] M. Littmann, N. Bordin, M. Heinzinger, K. Schütze, C. Dallago, C. Orengo, and B. Rost, ''Clustering FunFams using sequence embeddings improves EC purity,'' *Bioinformatics*, vol. 37, no. 20, pp. 3449–3455, Oct. 2021, doi: 10.1093/bioinformatics/btab371.

[44] J. Singh, T. Litfin, J. Singh, K. Paliwal, and Y. Zhou, ''SPOT-contact-LM: Improving single-sequence-based prediction of protein contact map using a transformer language model,'' *Bioinformatics*, vol. 38, no. 7, pp. 1888–1894, Mar. 2022, doi: 10.1093/bioinformatics/btac053.

[45] S. Mahbub and M. S. Bayzid, ''EGRET: Edge aggregated graph attention networks and transfer learning improve protein–protein interaction site prediction,'' *Briefings Bioinf.*, vol. 23, no. 2, Mar. 2022, doi: 10.1093/bib/bbab578.

[46] Y. Li, G. B. Golding, and L. Ilie, ''DELPHI: Accurate deep ensemble model for protein interaction sites prediction,'' *Bioinformatics*, vol. 37, no. 7, pp. 896–904, May 2021, doi: 10.1093/bioinformatics/btaa750.

[47] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, and R. Fergus, ''Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences,'' *Proc. Nat. Acad. Sci. USA*, vol. 118, no. 15, 2021, Art. no. e2016239118, doi: 10.1073/pnas.2016239118.

[48] M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, and B. Rost, ''Modeling aspects of the language of life through transfer-learning protein sequences,'' *BMC Bioinf.*, vol. 20, no. 1, p. 723, Dec. 2019, doi: 10.1186/s12859-019-3220-8.

[49] E. Asgari and M. R. K. Mofrad, ''Continuous distributed representation of biological sequences for deep proteomics and genomics,'' *PLoS ONE*, vol. 10, no. 11, Nov. 2015, Art. no. e0141287, doi: 10.1371/journal.pone.0141287.

[50] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, and A. Rives, ''Language models of protein sequences at the scale of evolution enable accurate structure prediction,'' *bioRxiv*, 2022, doi: 10.1101/2022.07.20.500902.

[51] O. Barukab, F. Ali, W. Alghamdi, Y. Bassam, and S. A. Khan, ''DBP-CNN: Deep learning-based prediction of DNA-binding proteins by coupling discrete cosine transform with two-dimensional convolutional neural network,'' *Expert Syst. Appl.*, vol. 197, Jul. 2022, Art. no. 116729, doi: 10.1016/j.eswa.2022.116729.

[52] C. Zou, J. Gong, and H. Li, ''An improved sequence based prediction protocol for DNA-binding proteins using SVM and comprehensive feature analysis,'' *BMC Bioinf.*, vol. 14, no. 1, p. 90, Dec. 2013, doi: 10.1186/1471-2105-14-90.

[53] X. Ma, J. Guo, and X. Sun, ''DNABP: Identification of DNA-binding proteins based on feature selection using a random forest and predicting binding residues,'' *PLoS ONE*, vol. 11, no. 12, Dec. 2016, Art. no. e0167345, doi: 10.1371/journal.pone.0167345.

[54] R. Zaman, S. Y. Chowdhury, M. A. Rashid, A. Sharma, A. Dehzangi, and S. Shatabda, ''HMMBinder: DNA-binding protein prediction using HMM profile based features,'' *Biomed Res Int*, vol. 2017, Nov. 2017, Art. no. 4590609, doi: 10.1155/2017/4590609.

[55] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, ''The protein data bank, 1999-,'' Int. Tables Crystallogr., Int. Union Crystallogr., Chester, U.K., Tech. Rep., 2006, pp. 675–684, doi: 10.1107/97809553602060000722.

[56] S. Altschul, ''Gapped BLAST and PSI-BLAST: A new generation of protein database search programs,'' *Nucleic Acids Res.*, vol. 25, no. 17, pp. 3389–3402, Sep. 1997, doi: 10.1093/nar/25.17.3389.

[57] J. A. Ruffolo and J. J. Gray, ''Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies,'' *Biophys. J.*, vol. 121, no. 3, pp. 155a–156a, Feb. 2022, doi: 10.1016/j.bpj.2021.11.1942.

[58] R. Chowdhury, N. Bouatta, S. Biswas, C. Floristean, A. Kharkar, K. Roy, C. Rochereau, G. Ahdritz, J. Zhang, G. M. Church, P. K. Sorger, and M. AlQuraishi, ''Single-sequence protein structure prediction using a language model and deep learning,'' *Nature Biotechnol.*, vol. 40, no. 11, pp. 1617–1623, Nov. 2022, doi: 10.1038/s41587-022-01432-w.

[59] B. Hie, E. D. Zhong, B. Berger, and B. Bryson, "Learning the language of viral evolution and escape," *Science*, vol. 371, no. 6526, pp. 284–288, Jan. 2021, doi: 10.1126/science.abd7331.

[60] N. Brandes, D. Ofer, Y. Peleg, N. Rappoport, and M. Linial, "ProteinBERT: A universal deep-learning model of protein sequence and function," *Bioinformatics*, vol. 38, no. 8, pp. 2102–2110, Apr. 2022, doi: 10.1093/bioinformatics/btac020.

[61] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," Oct. 2018, *arXiv:1810.04805*, doi: 10.48550/arxiv.1810.04805.

[62] J. Meier, R. Rao, R. Verkuil, J. Liu, T. Sercu, and A. Rives, "Language models enable zero-shot prediction of the effects of mutations on protein function," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, J. W. Vaughan, Eds. Red Hook, NY, USA: Curran Associates, 2021, pp. 29287–29303. [Online]. Available: https://proceedings.neurips.cc/paper/2021/file/f51338d736f95dd42427296047067694-Paper.pdf

[63] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems*, I. Guyon, U. von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/2cad8fa47bbef282badbb8de5374b894-Paper.pdf

[64] G. E. Hinton, S. Sabour, and N. Frosst, "Matrix capsules with EM routing," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, vol. 115, 2018.

[65] D. Pan, Y. Lu, and P. Kang, "A deep learning model for multi-label classification using capsule networks," in *Proc. Int. Conf. Intell. Comput.*, 2019, pp. 144–155, doi: 10.1007/978-3-030-26763-6_14.

[66] K. K. Kumar, G. Pugalenthi, and P. N. Suganthan, "DNA-prot: Identification of DNA binding proteins from protein sequence information using random forest," *J. Biomolecular Struct. Dyn.*, vol. 26, no. 6, pp. 679–686, Jun. 2009, doi: 10.1080/07391102.2009.10507281.

[67] B. Liu, J. Xu, X. Lan, R. Xu, J. Zhou, X. Wang, and K.-C. Chou, "IDNA-Prot|dis: |IDNA-Prot|dis: Identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition," *PLoS ONE*, vol. 9, no. 9, Sep. 2014, Art. no. e106691, doi: 10.1371/journal.pone.0106691.

[68] S. Zhao, Y. Ding, X. Liu, and X. Su, "HKAM-MKM: A hybrid kernel alignment maximization-based multiple kernel model for identifying DNA-binding proteins," *Comput. Biol. Med.*, vol. 145, Jun. 2022, Art. no. 105395, doi: 10.1016/j.compbiomed.2022.105395.

[69] Y. Zou, Y. Ding, J. Tang, F. Guo, and L. Peng, "FKRR-MVSF: A fuzzy kernel ridge regression model for identifying DNA-binding proteins by multi-view sequence features via Chou's five-step rule," *Int. J. Mol. Sci.*, vol. 20, no. 17, p. 4175, Aug. 2019, doi: 10.3390/ijms20174175.

[70] M. K. Patrick, A. F. Adekoya, A. A. Mighty, and B. Y. Edward, "Capsule networks—A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 1, pp. 1295–1310, Jan. 2022, doi: 10.1016/j.jksuci.2019.09.014.

[71] D. N. Ezechukwu and Y. L. Moullec, "CapsNet on embedded devices in a data scarce scenario," in *Proc. 18th Biennial Baltic Electron. Conf. (BEC)*, Oct. 2022, pp. 1–6, doi: 10.1109/BEC56180.2022.9935600.

[72] Y. Wang, B. Wang, J. Jiang, J. Guo, J. Lai, X.-Y. Lian, and J. Wu, "Multitask CapsNet: An imbalanced data deep learning method for predicting toxicants," *ACS Omega*, vol. 6, no. 40, pp. 26545–26555, Oct. 2021, doi: 10.1021/acsomega.1c03842.

MUHAMMAD KHAERUL NAIM received the bachelor's degree in electrical engineering from the Sepuluh Nopember Institute of Technology, Indonesia, in 2011, and the master's degree in electrical engineering from the University of Indonesia, Indonesia, in 2016. He is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Informatics. His main areas of research interests include image processing, bioinformatics, computer vision, and deep learning.

TATI RAJAB MENGKO received the Ir.B.S. degree in electrical engineering from the Bandung Institute of Technology (ITB), Bandung, Indonesia, in 1977, and the Dr.Eng. degree from ENSERG-INPG-Grenoble, France, in 1985. She has been with the School of Electrical Engineering and Informatics, ITB, since 1978. In 2006, she was an Image Processing Professor with the School of Electrical Engineering and Informatics, ITB, where she is currently the Head of the Biomedical Engineering Research Division. Her research interests include image processing and biomedical engineering instrumentation. She received the ITB Innovation Award, in 2015, for her contribution to the development of a non-invasive vascular analyzer device. She has presided over a multitude of conferences, including the International Conference on Instrumentation, Communication, Information Technology, and Biomedical Engineering (ICICI-BME).

RUKMAN HERTADI received the Ph.D. degree from the Tokyo Institute of Technology, in 2003. He is Professor of physical biochemistry with the Department of Chemistry, Bandung Institute of Technology, Bandung, Indonesia, where he is the Head of the Biomaterials and Biophysics Research Group. His research interests include computational biochemistry, biomaterials, and protein stability and dynamics.

AYU PURWARIANTI received the Ph.D. degree from the Toyohashi University of Technology, in December 2007, with dissertation title of "Cross Lingual Question Answering System (Indonesian Monolingual QA, Indonesian-English CLQA, and Indonesian-Japanese CLQA)." The dissertation was in the field of natural language processing, also known as computational linguistics, which is part of the knowledge domain of artificial intelligence. Since then, she has been a Lecturer with the Bandung Institute of Technology (ITB). Since August 2019, she has been the Chair of the Artificial Intelligence Centre, ITB. In addition to teaching and conducting research, she was the Chair of the Indonesian Association for Computational Linguistics, from 2016 to 2018. From 2017 to 2019, she also served as the Chair for the IEEE Education Chapter of the Indonesian Section. Since 2015, she has been with IABEE. In 2018, she established the company Prosa.ai.

MEREDITA SUSANTY received the master's degree in management of information technology from the University of Nottingham, in 2015. She is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Informatics, University of Pertamina. She is also with the Department of Computer Science, University of Pertamina. Her main areas of research interests include NLP, bioinformatics, machine learning, and deep learning.

• • •