Trends in Biotechnology

CellPress

Review

Enzyme functional classification using artificial intelligence

Ha Rim Kim[1,2,6], Hongkeun Ji[1,2,6], Gi Bae Kim[1,2,4], and Sang Yup Lee[1,2,3,4,5,7,*,@]

Enzymes are essential for cellular metabolism, and elucidating their functions is critical for advancing biochemical research. However, experimental methods are often time consuming and resource intensive. To address this, significant efforts have been directed toward applying artificial intelligence (AI) to enzyme function prediction, enabling high-throughput and scalable approaches. In this review, we discuss advances in AI-driven enzyme functional annotation, transitioning from traditional machine learning (ML) methods to state-of-the-art deep learning approaches. We highlight how deep learning enables models to automatically extract features from raw data without manual intervention, leading to enhanced performance. Finally, we discuss the discovery of novel enzyme functions and generation of *de novo* enzymes through the integration of generative AIs and bio big data as future research directions.

Highlights

Recent advances in computational biology, especially deep learning models, have significantly accelerated the discovery and annotation of enzyme functions.

Diverse enzyme feature extraction methods and machine learning algorithms have facilitated the development of high-performing computational models for enzyme function classification.

Deep neural network architectures, including convolutional neural networks, recurrent neural networks, transformers, and graph neural networks, have revolutionized data-driven approaches in predicting enzyme functions.

Artificial intelligence has transformed enzyme function classification, making it more scalable and accurate than ever before.

The necessity of AI-based enzyme functional classification

Metabolism lies at the core of cellular function, orchestrating a series of biochemical reactions to support growth, maintenance, and adaptation. These reactions are organized into a metabolic network, in which enzymes act as catalysts at each step. A well-characterized metabolic network enables researchers to understand how nutrients are processed, how energy is generated, and how cells respond to environmental changes. Thus, unveiling the functions of enzymes is fundamental for biochemical research. While the advent of whole-genome sequencing has revolutionized our ability to uncover the genetic basis of metabolism by identifying genes potentially encoding enzymes, accurately annotating functions to these genes remains an ongoing challenge. Although traditional experimental approaches, such as *in vitro* enzyme assays, are crucial for verifying specific enzyme functions, they are challenging to conduct on a large, high-throughput scale. Consequently, a considerable number of enzymes remain functionally uncharacterized, creating a significant gap between genome-sequencing achievements and functional annotation.

To address this gap, computational approaches have emerged as powerful tools for rapidly inferring enzyme functions [1,2]. These computational approaches complement experimental assays by narrowing down candidate enzymes for targeted validation, thereby accelerating the overall process of enzyme function prediction. Although such methods have proven effective, they rely heavily on existing annotated databases, which limit their applicability for enzymes with few known homologs. Recent advances in graphics processing units, computational algorithms, and the continual expansion of biological data have enabled the development of **deep learning** (see Glossary) models capable of recognizing complex patterns in protein sequences [3,4]. As a result, these models can predict enzyme functions with unprecedented accuracy and scalability.

In this review, we highlight advances in computational biology for enzyme function prediction, ranging from conventional **machine learning (ML)** models to state-of-the-art deep learning

[1]Metabolic and Biomolecular Engineering National Research Laboratory, Department of Chemical and Biomolecular Engineering (BK21 four), KAIST Institute for BioCentury, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea
[2]Systems Metabolic Engineering and Systems Healthcare Cross-Generation Collaborative Laboratory, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea
[3]Graduate School of Engineering Biology, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea
[4]BioProcess Engineering Research Center, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea
[5]Center for Synthetic Biology, Korea Advanced Institute of Science and Technology (KAIST), Daejeon 34141, Republic of Korea

models. By doing so, we provide a comprehensive overview of the field, highlighting key concepts, methods, and ongoing challenges in bridging the gap between sequence data and functional annotation.

Predicting enzyme functions using machine learning

Although sequence similarity-based approaches have been the predominant method for enzyme functional classification, their effectiveness in predicting enzyme functions is constrained by convergent and divergent evolutionary processes. Convergent evolution can cause proteins with similar enzyme functions to have low sequence similarity, whereas divergent evolution can result in proteins with different functions to have high sequence similarity [5]. To overcome the limitations of sequence similarity-based approaches, recent bioinformatics studies have increasingly adopted an ML for enzyme functional classification. An ML is a subfield of **artificial intelligence (AI)** that improves task performance through learning from data [6]. This data-driven approach identifies inherent data patterns without explicit programming, enabling the prediction of enzyme functions beyond sequence similarity.

Since the performance of an ML model is significantly affected by the types of input representation, diverse feature extraction methods should be explored. Feature extraction involves processing raw data to extract meaningful information for model training and inference. Effective feature extraction requires selecting information that best captures the characteristics of the data, informed by a deep understanding of the relevant field. Similarly, it is crucial to select an appropriate ML algorithm and design the model structure based on the characteristics of the task and the input representation. Here, we introduce various feature extraction techniques and ML algorithms commonly applied in enzyme function prediction (Figure 1).

Enzyme feature extraction

Amino acid composition (AAC) [7] is one of the most widely used features for representing enzymes in numerical data, quantifying the frequency of each amino acid within the entire sequence (Figure 1A). Given that each amino acid has specific physicochemical properties, AAC provides a concise summary of protein characteristics and is used extensively for enzyme functional classification. For instance, a **k-nearest neighbor (kNN)**-based classification model was developed to predict enzyme classes by using AAC of a protein sequence as input [8]. Similarly, AAC was used as input for a **support vector machine (SVM)** alongside physicochemical properties to predict enzyme classes [9].

Various physicochemical properties derived from amino acids, such as molecular weight, isoelectric point, atomic composition, and number of charged residues, reveal critical aspects of protein structural and functional characteristics (Figure 1A). These properties provide complementary features to AAC, being frequently used in enzyme classification studies along with AAC. Several computational feature extraction tools have been developed to analyze and vectorize the physicochemical properties within enzyme sequences [10–12]. These calculated features have proven useful for enzyme functional classification when integrated with ML models. For example, ECPred uses Pepstats to extract input protein features for training an SVM that predicts the **Enzyme Commission number (EC number)** of the protein [13]. Considering the numerous types of physicochemical feature that can be applied to feature extraction, using all of them can lead to an excessively high-dimensional feature vector, triggering the **curse of dimensionality**. To address this, dimension reduction methods, such as principal component analysis (PCA), can be applied [14].

Higher-order structures of proteins (i.e., secondary, tertiary, and quaternary structures) are more directly correlated with biological functions than are sequence-based representations of amino

[6]These authors equally contributed to this work
[7]Website: https://mbel.kaist.ac.kr

*Correspondence:
leesy@kaist.ac.kr (S.Y. Lee).
[@]X: @mbelmbel99

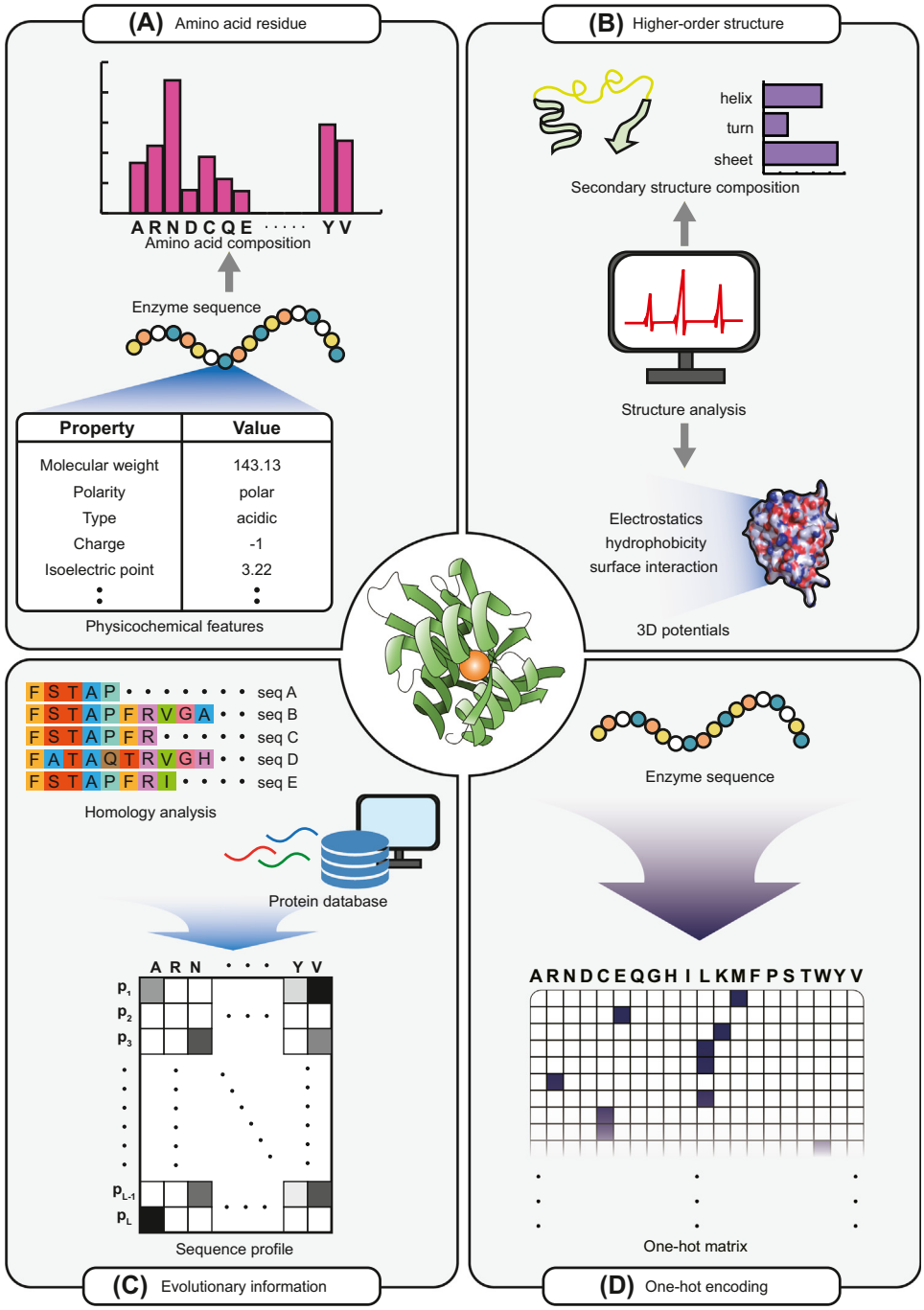

Figure 1. Schematic of the types of enzyme feature that can be elucidated with a machine learning model. (A) Extraction of the amino acid composition (AAC) and physicochemical properties of each amino acid from the enzyme sequence. While AAC represents the frequency of each amino acid residue in the entire enzyme sequence, the physicochemical properties of amino acids provide insights into the structural and functional characteristics of proteins. (B) Extraction of higher-order structure features from protein structure analysis. The secondary and tertiary structure of proteins are more directly correlated with their biological functions than is the primary structure. (C) Extraction of

(Figure legend continued at the bottom of the next page.)

Glossary

Artificial intelligence (AI): technology that mimics human cognitive abilities to perform learning, reasoning, problem-solving, and decision-making. It includes an ML and deep learning.

Artificial neural network (ANN): an ML model comprising artificial neurons (nodes), synapses (edges), and layers; the learning process is conducted by gradually optimizing the weights assigned to each neuron through a signal transmission process called backpropagation. Activation functions in neural network layers can convert the linear signals of each layer into nonlinear signals.

Contrastive learning: a kind of self-supervised learning that trains models by comparing positive and negative sample pairs, maximizing the similarity of related samples while minimizing that of unrelated ones to learn useful representations without explicit labels.

Convolutional neural network (CNN): a deep learning architecture that processes grid-like data through convolutional layers that extract local patterns, making them effective for tasks such as image and video recognition, natural language processing, and other structured data applications.

Curse of dimensionality: a phenomenon in an ML where increasing the number of features (dimensions) leads to inefficient model training and reduced performance due to data sparsity and increased computational complexity.

Decision tree: an ML algorithm that follows a tree-structured approach, recursively splitting data based on feature values to maximize the homogeneity of leaf nodes, improving classification or regression accuracy.

Deep learning: a subfield of an ML that uses multilayered neural networks to learn complex and nonlinear patterns in data. It automatically extracts features without rigorous feature selection process, and typically requires large data sets for training.

Enzyme Commission number (EC number): an internationally recognized numerical classification system that systematically categorizes enzymes based on their catalytic reaction type. It comprises four digits arranged in a hierarchical manner, providing increasing specificity, and is denoted in the format 'EC: X.X.X.X'.

acids. Therefore, structural information can also be used for enzyme functional classification (Figure 1B). The composition of each secondary structure, including helices, turns, and sheets, serves as a representative feature describing the local conformation of polypeptide backbones. Bioinformatic tools, such as STRIDE [15] and PSIPRED [16], have been developed to calculate secondary structure compositions from raw sequences, enabling enzyme functional classification using ML models and secondary structural information. For instance, SVM models trained using secondary structure compositions derived from STRIDE were used to predict enzyme classes, alongside other structural (e.g., surface area, disulfide bonds, and AAC on the surface) and physicochemical (e.g., AAC, types of cofactor, and metal ions) features [17]. While secondary structures elucidate general folding patterns, they provide limited insight into the specific biological functions of proteins. Studies have also explored features related to whole 3D structures; for example, electrostatic, hydrophobic, and surface potentials can elucidate how each amino acid interacts within the protein structure. These features have been used as inputs of linear discriminant analysis, SVM, and kNN to predict enzyme classes [18].

Furthermore, some studies co-utilized evolutionary information, such as sequence profiles and homology scores, with intrinsic information mentioned earlier (Figure 1C). For example, EzyPred integrates a **position-specific scoring matrix (PSSM)** along with functional domain composition for enzyme function prediction to effectively capture the information of evolutionarily conserved regions in protein sequences [19]. Similarly, ECPred predicts EC numbers using three modules: (i) subsequence profile mapping, which uses the PSSM calculated from *k*-mers of the protein sequence; (ii) kNN for processing BLAST scores, based on the enzyme family to which the sequences with the highest homology sequences belong; and (iii) SVM for processing physicochemical features derived from Pepstats [13].

General machine learning algorithms for classification

The selection of suitable ML algorithms is crucial for effectively capturing the intrinsic patterns in protein representations. Since each algorithm has distinct advantages and limitations, it is necessary to select appropriate algorithms based on the characteristics of the training data and the learning objectives. kNN is an instance-based learning algorithm that makes predictions from existing data rather than optimizing explicit models (Figure 2A) [20]. It has been applied to predict enzyme functions by assigning the most common functional class among the nearest neighbors in the same feature space. For example, EnzML, a multilabel kNN classifier, was trained on functional domain composition to predict EC numbers [21]. kNN-based algorithms offer a highly intuitive training process and facilitate the straightforward identification of data instances that significantly influence predictions. However, kNN computes pairwise distances for every query against all training data and requires storing the full dataset, resulting in very high computational costs and hindering high-throughput analysis of expanding whole-genome sequencing data.

ML algorithms that use decision boundaries have addressed the computational inefficiency of kNN-based methods by designating specific regions of the feature space for each class. SVM is a prominent example of such boundary-based classification (Figure 2B) [22]. SVM identifies a hyperplane that maximizes the margin, which is the distance between the hyperplane and the closest data points from each class, performing well even for high-dimensional input data. Due to this advantage, SVM is one of the most frequently used algorithms in enzyme function

Graph neural network (GNN): a deep learning architecture designed to process graph-structured data. It propagates and aggregates information among nodes to learn representations that capture both individual node (or edge) features and overall graph features.
k-nearest neighbor (kNN): an ML algorithm that predicts outcomes by analyzing the *k*-closest data points in the feature space, using distance metrics, such as Euclidean distance, to determine similarity.
Large language model (LLM): a deep neural network with billions of parameters trained on vast text corpora. It captures complex linguistic patterns to generate contextually coherent text, often serving as the foundation for various natural language-processing tasks.
Machine learning (ML): a subfield of AI that enables systems to learn patterns from data and improve performance without explicit programming.
Position-specific scoring matrix (PSSM): a type of evolutionary feature that quantifies the probability of each amino acid occurring at a specific position within a protein sequence. It represents conservation and variation of sequences.
Random forest: an ML algorithm that builds an ensemble of decision trees, each trained on a randomly sampled subset of data and features. Predictions are made by averaging or voting across trees.
Recurrent neural network (RNN): a deep learning architecture designed for sequential data by recursively updating a hidden state that encodes past information. This mechanism allows RNN to capture temporal dependencies.
Supervised learning: an ML paradigm that trains models using labeled data sets to map inputs to corresponding outputs. The learning process involves minimizing the discrepancy between predicted and true values.
Support vector machine (SVM): an ML algorithm that finds an optimal hyperplane to maximize the margin between classes in a feature space. It supports both linear and non-linear classification through the use of kernel functions.
Transformer: a deep learning architecture that uses self-attention mechanisms to process entire sequences in parallel, thereby modeling contextual relationships between all elements simultaneously.

evolutionary information by homology analysis. (D) One-hot encoding representation of a protein sequence. The single-letter amino acid code of a protein sequence is transformed into a one-hot encoded representation in the form of a 2D matrix. In this matrix, the rows correspond to each position in the sequence, and the columns represent each amino acid. All elements are zero except for the positions corresponding to each residue, which are set to one.

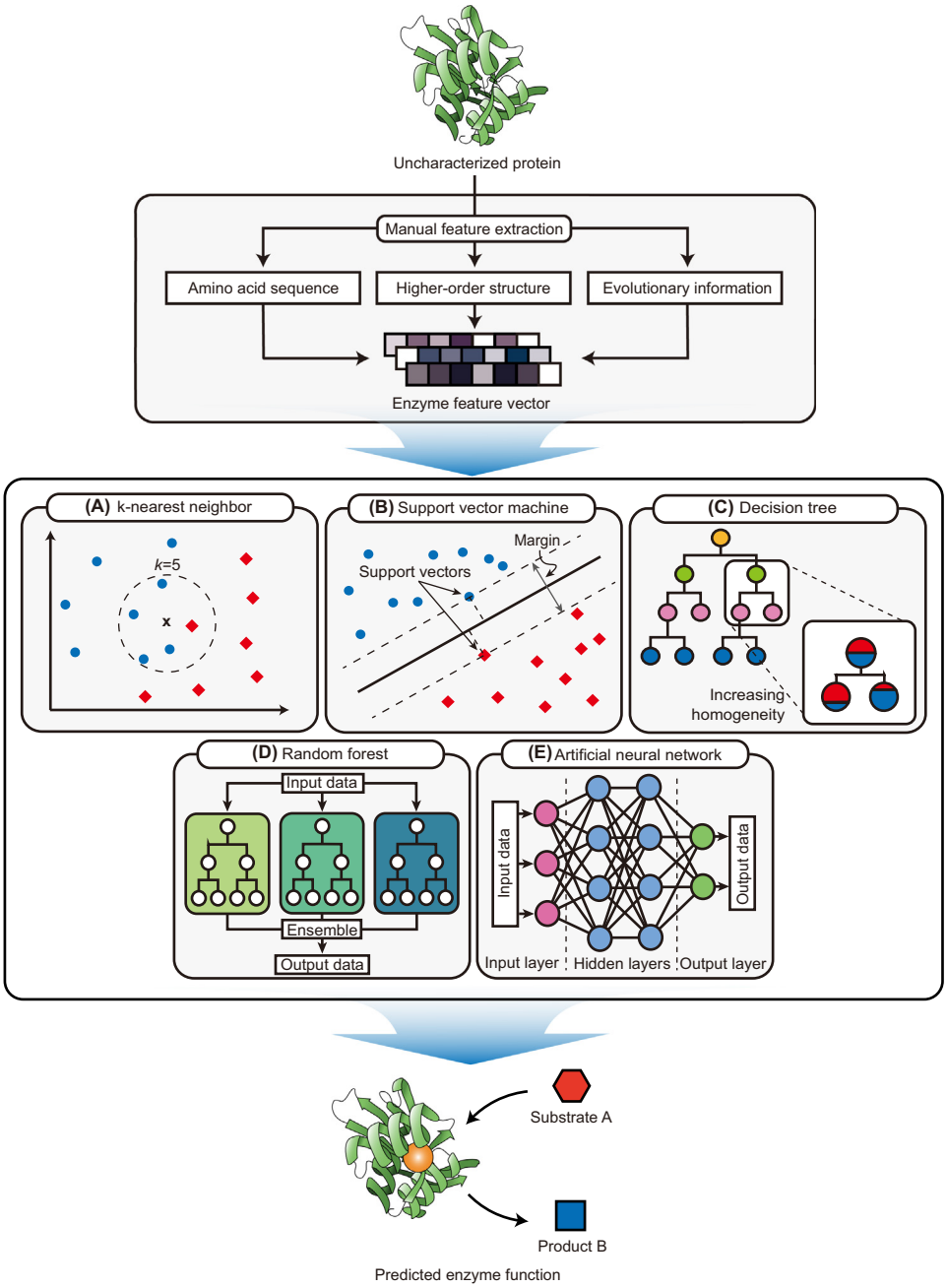

Figure 2. Schematic of machine learning (ML) algorithms frequently used for enzyme function prediction. The top panel illustrates that the various enzyme features illustrated in Figure 1 serve as inputs for an ML-based enzyme function prediction. (A) Classification process of k-nearest neighbor (kNN). kNN predicts classes by calculating distances in feature space and selecting the *k*-nearest neighbors. (B) Classification process of the support vector machine (SVM). SVM optimizes the decision boundary by maximizing the distance between the nearest data points of each class (support vectors) and the hyperplane. (C) Classification process of decision trees. The decision tree utilizes a tree-like structure where nodes split data to maximize subset homogeneity, defined as the degree to which a set comprises exclusively data with the same attributes. (D) Classification process of the random forest. Random forest is an ensemble learning method that builds multiple decision trees during training and combines their outputs to improve performance and reduce overfitting.

(Figure legend continued at the bottom of the next page.)

prediction tasks. For example, SVM-Prot was developed to predict EC numbers and protein families of input proteins using SVM [23]. Separate SVM models were trained for each functional class (i.e., EC numbers and protein families) using protein feature vectors that integrate various properties, such as AAC, secondary structures, and other physicochemical features.

The major challenges in using decision boundary-based algorithms are their sensitivity to feature distribution and the difficulty in interpreting their prediction results. **Decision tree** algorithms address these issues by recursively splitting the data at each node based on input features, offering inherent robustness to feature scaling and a more interpretable model structure (Figure 2C) [24]. For example, EFICAz^2 uses a decision tree to integrate prediction outputs from various modules (e.g., SVM-based and functionally discriminating residue-based modules), thereby improving EC number classification performance [25]. **Random forest**, an ensemble of multiple decision trees, has also been used for enzyme functional classification (Figure 2D) [26]. PredictEFC was developed to predict enzyme classes using random forests, with functional domain information retrieved from the InterPro database as inputs [27]. PredictEFC showed higher performance compared with an SVM-based model trained on the same dataset, highlighting the effectiveness of random forests in enzyme functional classification.

Artificial neural networks (ANNs) are ML algorithms inspired by the transmission of information between neurons via synapses in the human brain (Figure 2E) [28]. For decades, ANNs have been used to capture complex and nonlinear data patterns, enabling the analysis of nonlinear relationships between protein sequences and their function. For instance, ProtFun was developed as an ensemble of ANNs to predict enzyme functional classes from various physicochemical properties, including hydrophobicity, number of charged residues, and secondary structures [29]. However, in the early days of ANNs, computational capabilities were limited and, thus, only shallow network architectures were utilized.

As discussed earlier, an ML-based approaches should be carefully designed with well-chosen representations of enzymes. Such preparation demands substantial preprocessing time and an in-depth understanding of enzymology related to catalytic reactions. However, manually curated features often fail to capture the inherent complexity in biological data, limiting the ability of AI to accurately learn enzyme functional characteristics. Furthermore, traditional ML models are insufficient to model the intricate, complex relationships in biological data, shaped and evolved over billions of years. In summary, the difficulty in manually extracting optimal features and the challenge of unraveling complex nonlinear relationships between data features and labels have been major bottlenecks in traditional ML-based predictions.

Predicting enzyme functions using deep learning

Recent advances in deep learning, especially the development of various neural network architectures, have significantly enhanced the performance of enzyme function prediction. By leveraging deep and complex network architectures, deep learning models inherently learn intricate patterns directly from raw data (e.g., amino acid sequences) without the need for extensive feature extraction [3]. Here, we examine the progression of deep neural network architectures and their role in enabling high-performance prediction of protein functions, specifically in terms of EC numbers and Gene Ontology (GO) terms (Table 1).

(E) Classification process of the artificial neural network (ANN). ANN comprises layers of artificial neurons, the weights of which are updated through backpropagation. Activation functions, such as sigmoid or rectified linear unit, enable the model to learn nonlinear data patterns. By leveraging these models, catalytic activities and specific functions of enzymes can be predicted based on their intrinsic properties.

Table 1. Deep learning-based tools for predicting EC numbers and GO terms[a]

Tool	Output type[b]	Input type	Main neural network architecture	Characteristics	Refs
DeepEC	EC number (4669 EC numbers)	Protein sequence	Convolutional layers	Combination with homologous search module	[35]
DEEPre	EC number (3517 EC numbers)	Protein sequence, PSSM, solvent accessibility, secondary structure, functional domains	Convolutional layers, LSTM layers	Hierarchical classification using models specialized in each EC number	[32]
mIDEEPre	EC number (3517 EC numbers)	Protein sequence, PSSM, functional domains	Convolutional layers	Hierarchical classification with DEEPre after second digit Multi-label classification based on first digit of enzyme	[33]
DCNN	EC number	Mutation features, distance features, angle features	Convolutional layer	Use of kNN algorithm for integrating each feature and inferring output	[34]
HECNet	EC number (402 EC numbers)	Protein sequence, PSSM, solvent accessibility, secondary structure, functional domains, disordered region, amino acid composition	Convolutional layer, LSTM layers	Use of Siamese network employing triplet loss	[36]
HDMLF	EC number	Protein sequence	Transformer layers, BiGRU layers, attention layers	Use of multiple sequence alignment to fine-tune prediction and multitask learning to optimize outputs predicted Prediction of promiscuous enzyme functions	[72]
UDSMProt	EC number (3978 EC numbers), GO term (5101 GO terms), remote homology detection, fold detection	Calculated features, protein sequence	LSTM layers	Pretraining of LSTM layers Fine-tuning of classifier	[61]
DeepECtransformer	EC number (5360 EC numbers)	Protein sequence	Transformer layers, convolutional layers	Combination of homologous search module	[39]
PhiGnet	EC number, GO term	Protein sequence	Transformer layers, graph convolutional layers	Use of evolutionary couplings and residue community information	[54]
CLEAN	EC number (5242 EC numbers)	Protein sequence	Transformer layers	Prediction of EC numbers using contrastive learning	[55]
MAPred	EC number (5242 EC numbers)	Protein sequence	Transformer layers, convolutional layers, cross- attention layers	Use of embeddings extracted by ESM-1b and ProstT5, and of global and local features	[73]
GraphEC	EC number (5106 EC numbers), active site, optimal pH	Protein sequence	Transformer layers, graph convolutional network	Use of protein language model, ESMFold to extract structural features, and label diffusion to reflect homolog information	[59]
GearNet	EC number (538 EC numbers), GO term	Protein sequence, protein structure	Graph convolutional layers, edge message-passing layers	Use of three kinds of edge type and multiview contrastive learning for pretraining	[49]
DeeProtGO	GO term (22 246 GO terms)	Amino acid trigram	Fully connected layers	Hierarchical classification of GO terms	[74]
DeepGOZero	GO term (31 081 GO terms)	InterPro binary features	Fully connected layers	Use of zero-shot learning Learning to embed proteins in space where GO axioms have been embedded by ELEmbeddings method	[75]

(continued on next page)

Table 1. (continued)

Tool	Output type[b]	Input type	Main neural network architecture	Characteristics	Refs
DeepGraphGO	GO term	Protein network, InterPro binary features	Graph convolutional layers	Integration in NetGO framework	[76]
CFAGO	GO term	Protein network, protein domain, subcellular location	Transformer layers	Pretraining using self-supervised learning Fine-tuning for GO term prediction task Use of protein–protein interaction networks	[77]
Chemical-SA-BiLSTM	GO term	Protein sequence, chemical properties	Self-attention layer, bidirectional LSTM layers	Specialization for proteins from grain	[78]
CaLM	GO term	DNA sequence	Transformer layers	Learning patterns of synonymous codon usage Pretraining on cDNA sequences	[68]
DeepGOPlus	GO term (5220 GO terms)	Protein sequence	Convolutional layers	Combination with sequence homology search	[79]
DeepGOA	GO term	Protein sequence	Convolutional layers, graph convolutional layers	Specialization for maize proteins	[80]
NetGO2	GO term	Protein sequence	BiLSTM layers	Use of learn-to-rank framework	[81]
ProteInfer	GO term (32 109 GO terms)	Protein sequence	Dilated convolutional layers	Provision of coarse-grained functional localization	[37]
NetGO3	GO term	Protein sequence	Transformer layers	Use of protein language and learn-to-rank framework	[82]
PFresGO	GO term (2752 GO terms)	Protein sequence	Transformer layers	Use of protein language model and GO graph hierarchy	[62]
ESM-S	GO term	Protein sequence	Transformer layers	Pretraining on remote homology detection task	[83]
DeepGO-SE	GO term	Protein sequence	Transformer layers	Learning to embed proteins in space where GO axioms have been embedded by ELEmbeddings method	[63]
HNetGO	GO term (1340 GO terms)	Protein sequence	Graph attention network	Use of SeqVec to extract learned embeddings, protein–protein interaction networks, and GO graph hierarchy	[84]
DeepGO	GO term (27 760 GO terms)	Protein sequence	Convolutional layers	Use of protein–protein interaction networks and GO graph hierarchy	[85]
GAT-GO	GO term (2752 GO terms)	Protein sequence	Convolutional layers, graph attention networks	Use of protein structure information and protein language models	[86]
DeepFRI	EC number, GO term (2752 GO terms)	Protein sequence, protein structure	LSTM layers, graph convolutional layers	Use of protein structure information	[64]
TransFun	GO term (4921 GO terms)	Protein sequence, protein structure	Transformer layers, equivariant GNNs	Combination with sequence homology search Use of protein structure information	[66]
PredGO	GO term (5220 GO terms)	Protein sequence, protein structure	Transformer layers, geometric vector perceptron GNNs	Use of protein–protein interaction networks and protein structure information	[87]

[a]Abbreviations: EC, Enzyme Commission; GNN, graph neural network; GO, Gene Ontology; LSTM, long short-term memory.
[b]Numbers in parentheses indicate the number of EC numbers or GO terms covered by the tools when available.

Deep learning-aided prediction of EC numbers

The EC number is a nomenclature scheme for classifying enzymes based on the functions they perform [30]. It comprises four digits arranged in a hierarchical system. The first digit, known as the main class, is assigned based on the primary type of reactions the enzyme catalyzes: oxidoreductase (EC:1), transferase (EC:2), hydrolase (EC:3), lyase (EC:4), isomerase (EC:5), ligase (EC:6), and translocase (EC:7). Further classification into subclasses and sub-subclasses is determined by the second and third digits, respectively. These digits provide detailed information about the enzymatic reaction, specifying the type of compound, group, or bond involved at the catalytic site, with criteria varying across classes. The fourth digit indicates the substrate specificity of the enzyme in the catalyzed reaction. Given its clear description of enzyme functions, various deep learning models have been developed to predict EC numbers and classify enzyme functions (Figure 3, Key figure). Here, we discuss recent advances in deep learning used for predicting EC numbers. This knowledge provides valuable insights into bridging enzymology with computational science, potentially advancing the fields of biological sciences and biotechnology.

Deep learning-based enzyme functional classification has advanced with the development of various neural network architectures. **Convolutional neural networks (CNNs)** have been widely applied to enzyme sequences because their convolutional layers effectively extract local features through element-wise multiplication and summation between convolutional filters and inputs (Figure 3A) [31]. This capability allows CNNs to learn local contexts in enzyme sequences, such as functional motifs, resulting in high performance in enzyme functional classification. Various deep learning models using CNNs have been developed to predict EC numbers [32–37].

For example, DeepEC used one-hot encoding with three parallel convolutional layers using different filter sizes (Box 1 and Figure 1D) [35]. This approach enabled the neural networks to capture diverse local features, processed by max-pooling layers and fully connected layers to predict: (i) whether the input protein is an enzyme; (ii) EC numbers up to the third digit; and (iii) EC numbers up to the fourth digit. In cases where the neural networks cannot predict EC numbers (e.g., inconsistencies between predictions up to the third and fourth digits), homology analysis is conducted to predict EC numbers by assigning those of enzymes with high sequence similarity. DeepEC outperformed existing models (i.e., CatFam, DETECT v2, ECPred, EFICAz$^{2.5}$, and PRIAM) by achieving a precision of 0.920 and a recall of 0.455 on 201 enzyme sequences that had not been used to develop the tools, whereas the other EC number prediction tools exhibited comparably low precision and recall, ranging from 0.737 to 0.880 and 0.203 to 0.416, respectively.

Despite amino acid sequences being inherently sequential data rather than 2D representations, CNNs have shown promising results in enzyme function prediction. With advances in deep learning, **recurrent neural networks (RNNs)** designed for processing sequential data, particularly in natural language processing, are increasingly used for EC number prediction tasks using amino acid sequences (Figure 3B). Given their ability to extract more global contexts across protein sequences, deep learning models using RNNs combined with CNNs have been developed. For example, DEEPre uses neural networks comprising two modules: (i) a CNN followed by a long short-term memory (LSTM) network, a type of RNN; and (ii) multilayer perceptrons (MLPs) serving as classifiers [32]. Each module processes sequence length-dependent inputs (e.g., PSSM, solvent accessibility, secondary structure, and one-hot encoded protein representation) and sequence length-independent inputs (e.g., presence of functional domains). Subsequently, all features are concatenated to predict EC number digits through MLPs. DEEPre performs hierarchical EC number classification by first predicting whether the input protein is an enzyme, followed by sequential prediction of each digit of the corresponding EC number up to the third

Key figure

Schematic of deep learning algorithms for Enzyme Commission (EC) numbers and Gene Ontology (GO) term prediction

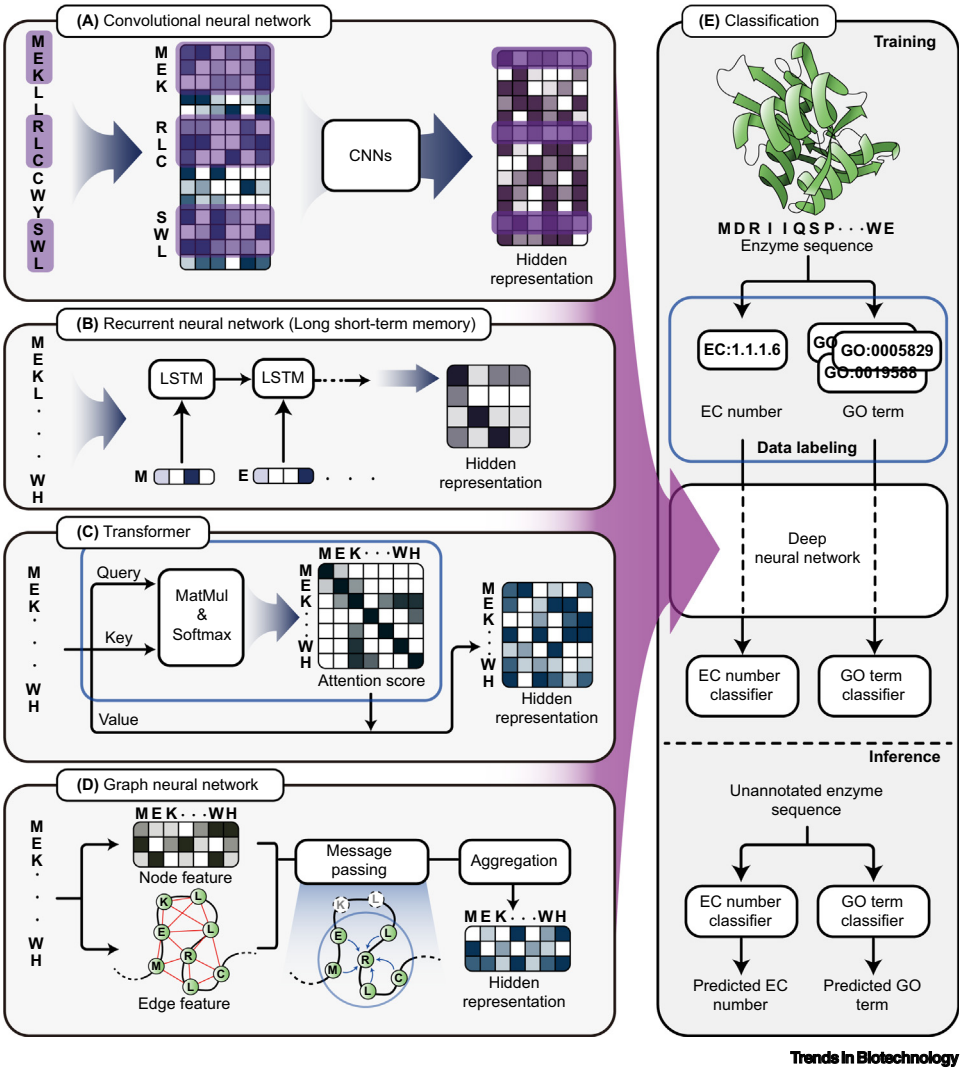

Figure 3. (A) Convolutional neural networks (CNNs) for learning local features of a protein sequence. CNNs capture local sequence patterns by scanning protein sequences with learnable filters, which aggregate local information into condensed embeddings. (B) Long short-term memory (LSTM), a type of recurrent neural network, for learning sequential features of a protein sequence. LSTMs process protein sequences sequentially by updating a hidden state at each residue. (C) Transformer for learning long-range relationships between all residues. Transformers use self-attention mechanisms to learn relationships between all residues in a protein sequence. They transform sequence embeddings into query, key, and value vectors, and compute attention scores using matrix multiplication (MatMul) and the Softmax function to generate context-aware representations that emphasize key interactions among residues. (D) Graph neural networks (GNNs) for learning structural information. GNNs model proteins as graphs where nodes represent residues (or atoms) and edges represent interactions defined by relationships, such as spatial proximity. Node (or edge) features are updated through the message-passing step, which propagates the aggregated information from neighboring nodes (or edges) to the target node (or edge), capturing both local and global structural contexts. (E) Scheme of enzyme function classification using deep learning. For training, enzyme sequences are labeled with targets (EC number or GO term), and the model learns to predict them using neural networks. A well-trained model can infer targets for unannotated enzyme sequences, even those absent from known data sets.

Box 1. Common methods of extracting protein features for deep learning models

Protein sequences are commonly represented using the single-letter amino acid code due to its simplicity and readability. However, this format consists solely of letters, which requires preprocessing to transform it into a machine-readable format suitable for deep learning models. Therefore, extensive efforts have focused on extracting rich information from protein sequences.

One-hot encoding is a conventional approach for handling categorical data, such as amino acids. It embeds a protein sequence into a 2D matrix with dimensions of sequence length multiplied by the number of amino acid types. Each amino acid is encoded into a vector, where all elements are zero except for the position corresponding to the amino acid, which is set to one. These encoded vectors are then stacked to represent the entire protein sequence. In addition to sequence-derived representations, deep learning models also utilize other inputs, such as physicochemical properties, probabilistic and statistical models, and evolutionary information, providing comprehensive information about input proteins [32,33,36,61].

Recently, protein representations extracted from pretrained models have become more commonly used as inputs for deep learning models because they capture various aspects of protein information. For instance, ESM-1b is a notable protein language model with 650 million parameters pretrained on 27 million protein sequences [43]. It uses masked language models specifically tailored for proteins, learning evolutionary relationships between residues, such as coevolution and contact. With advances in computational resources and the integration of larger data sets, ESM-1b has evolved into ESM2, offering more robust and evolutionarily rich representations [50].

Similar to the ESM models, ProtBERT and ProtT5 are also proficient in contextual information at the residue level using masked language models [45]. ProtBERT uses a BERT-based token masking approach, training on single amino acid masking to learn residue-level relationships, while ProtT5 uses span masking in an encoder–decoder structure to predict entire masked segments, capturing broader contextual patterns. Embeddings derived from such pretrained protein language models have demonstrated superior performance in downstream tasks, including classification, autoregression, and generation, compared with previous deep learning models [44,45,47]. The most recent development, ESM3, represents a state-of-the-art model trained on sequences, structures, and various properties of proteins [53]. This advance allows the protein language model to not only create representations from diverse aspects, but also function as a generative model for generating novel sequences, structures, and properties of proteins. Thus, the evolution of protein language models has significantly expanded the frontier of protein science.

digit. DEEPre was further upgraded to mlDEEPre, capable of predicting multilabel enzymes with multiple functions, such as promiscuous enzymes [33].

With the development of deep learning and the advent of **large language models (LLMs)** using **transformer** layers, the paradigm of protein function analysis has entered a new phase. Transformer layers are highly effective at learning relationships between residues, even when they are far apart (Figure 3C) [38]. DeepECtransformer uses a combination of transformer layers and convolutional layers to learn long-range interactions within amino acid residues [39]. An input protein sequence is fed into two consecutive transformer layers, followed by two convolutional layers, effectively learning both local and global features of the sequence. If an ANN in DeepECtransformer predicts no EC number for a given sequence, the homology search algorithm DIAMOND is used to assign an EC number based on homologous enzymes [40]. The use of transformer layers not only enhances the prediction performance of DeepECtransformer, but also enables the interpretation of the reasoning process of the deep learning model at the residue level, facilitating the analysis of enzyme functional domains. DeepECtransformer also surpassed DeepEC and DIAMOND on the test dataset derived from the UniProtKB/TrEMBL entries released in April 2018, showing macro and micro F1 scores of 0.8093 and 0.9611, respectively, which are 0.2703 and 0.2142 higher than the second-best scores, respectively [41].

To leverage the advantages of transformer architectures, which have enabled the development of BERT-like LLMs, protein language models (i.e., language models pretrained on numerous protein sequences) were constructed and pretrained on large amounts of protein sequences using masked language modeling (Box 1) [42–53]. Using embeddings extracted from such pretrained protein language models showed superior performance on downstream tasks compared with

previous deep learning models [44,45,47]. For example, PhiGnet used latent representations extracted from ESM-1b as inputs for two graph convolutional networks (GCNs) [43,54]. In PhiGnet, residue-level features were extracted from two graphs: (i) evolutionary couplings, which provide information on relationships between pairwise residues at two covariant sites; and (ii) residue communities, which contain hierarchical interactions among residues. By using evolutionary distances obtained from these two graphs as edges and ESM-1b-derived representations as node features, PhiGnet demonstrated superior performance in predicting EC numbers and GO terms. It also showed superior performance compared with other EC number prediction tools, such as BLAST, FunFams, DeepGO, DeepFRI, Pannzer, ProteInfer, and CLEAN, exhibiting an area under precision/recall curve (AUPR) of 0.89 and F_{max} scores of 0.88, which are 0.08 and 0.15 higher than those of the second-best models, respectively, on the test dataset.

Supervised learning, which trains an ML model to learn labels corresponding to inputs, has been the major approach in enzyme functional classification since the early stages of AI-based approaches. However, this training scheme restricts models to making predictions only among the labels presented in the training dataset, hindering their ability to generalize across all enzyme functions, especially for EC numbers with few data points. To address this issue, CLEAN, a model designed to cluster functionally similar enzymes in latent space, was developed [55]. CLEAN was trained using **contrastive learning**, where sequences annotated with the same EC number were drawn closer in latent space, while those with different EC numbers were pushed apart. This approach assigns EC numbers based on the closest enzymes in latent space to an input protein, resulting in superior performance in EC number prediction tasks compared with supervised learning-based methods. By leveraging the contrastive learning approach, CLEAN surpassed previous EC prediction tools, DeepEC, BLASTp, DEEPre, and ProteInfer, even showing a high accuracy of 0.8667 for fourth-level EC predictions on the halogenase dataset, while other tools achieved accuracies <0.4.

The remarkable advances in deep learning-based protein tertiary structure prediction tools, exemplified by AlphaFold and RoseTTAFold, have ushered in a new era of structure-based enzyme function analysis [56,57]. These tools have exponentially accumulated high-quality predicted structure data in protein structure database, such as AlphaFold DB[i], and their utilization of protein structural information has shown promise even in EC number prediction [58]. Recent deep learning models use not only protein representations derived from amino acid sequences, such as embeddings from protein language models, but also structural information derived from deep learning models pretrained on 3D protein structures.

With enhanced access to protein structure data, **graph neural networks (GNNs)** have been used to extract structural features for the functional classification of enzymes (Figure 3D). GraphEC, a deep learning-based model for predicting active sites, optimal pH, and EC numbers of input proteins, was developed using a GNN [50,59]. The nodes and edge features of the input graph are constructed using sequence embeddings from ProtTrans [45] and protein 3D structures predicted by ESMFold, a high-throughput protein structure prediction model [50]. By using these evolutionarily and structurally rich features and GNNs, GraphEC-AS is first trained to predict active sites of input proteins by assigning weight scores to each amino acid residue. These per-residue weighted scores are then used to predict EC numbers by emphasizing high-scoring residues to derive global features of the sequence. GraphEC also integrates label diffusion, a method that balances the final prediction between conserving the initial prediction and following homologs of the input protein, to predict EC numbers by considering homologous information. The integration of these approaches has enabled the effective use of structural information from deep learning and evolutionary information from homologous enzymes, resulting in the superior performance of GraphEC in predicting EC

numbers. On the NEW-392 and Price-149 datasets, which cover 177 and 56 different EC numbers, respectively, GraphEC achieved the highest F1 scores of 0.5910 and 0.6131, respectively among ECPred, GrAPFI, DeepEC, ProteInfer, ECPICK, and CLEAN, which ranged from 0.0927 to 0.4988 and from 0.0197 to 0.4947 [60].

Despite these efforts to predict EC numbers using deep learning, inherent challenges remain difficult to address: the scarcity of data for less-studied EC numbers and resulting dataset imbalances. These limitations result in lower performance when predicting EC numbers for enzymes with limited data. For example, while EC number prediction models have shown relatively high performance in predicting the first and second digits of EC numbers, performance tends to drop as predictions become more specific (e.g., predicting fourth-digit EC numbers). These challenges are inevitable when predicting enzyme functions in terms of EC numbers, but addressing them is crucial for achieving more accurate and robust enzyme functional classification (Figure 3E).

Deep learning-aided prediction of GO terms

As discussed, EC numbers provide a well-structured classification scheme for enzyme functions using four hierarchical digits. GO terms, which are controlled vocabularies of gene functionalities, also describe enzyme functions across three domains: molecular function (MF), cellular component (CC), and biological process (BP). As of November 2024, there were 40 635 GO terms[ii], providing more complex and nuanced descriptions of enzyme functions through multiple GO terms and their relationships. GO terms are structured as a directed acyclic graph, where each node represents a GO term and an edge denotes relationships between the GO terms. This complex hierarchy of GO terms makes predicting GO terms for a protein challenging.

Recent advances in deep learning models have facilitated extracting latent features directly from the raw representation of inputs (e.g., RGB values of an image) without requiring domain-specific feature engineering. As a result, GO term prediction models based solely on amino acid sequences have been developed. For example, ProteInfer, a deep learning model developed to predict EC numbers and GO terms, use deep dilated convolutional layers to increase the receptive field, facilitating the extraction of a more global context of the input sequence (Figure 3A) [37].

Advances in deep learning, especially in neural networks for sequential data, have significantly enhanced GO term prediction using amino acid sequences. For example, UDSMProt uses long short-term memory networks to predict EC numbers and GO terms (Figure 3B) [61]. The model was trained in two steps: pretraining on a large protein sequence database to learn a general understanding of proteins, followed by fine-tuning on task-specific datasets to predict enzyme function. Pretraining used a self-supervised autoregressive approach, where the neural network predicted the next amino acid residue in a sequence based on preceding amino acid residues. Fully connected layers were then added to the pretrained long short-term memory network and fine-tuned to predict GO terms for the input protein. UDSMProt showed superior AUPR scores of 0.472, 0.356, and 0.704 for MF, BP, and CC, respectively, among DiamondScore, DeepGO, and DeepGOCNN, on a dataset from Swiss-Prot annotations between January and October 2016, which had not been used to train benchmarking tools, thus allowing fair comparisons [41].

The development of transformer architecture has become a powerful approach in deep neural network development by using pretrained models for downstream processes through fine-tuning on task-specific datasets (Figure 3C). This strategy involves constructing LLMs using transformer-based neural networks and training them on large corpus data, which has been adopted in biotechnology, leading to the construction of protein language models. Since the

emergence of such models, deep learning models for the GO term prediction task have started to use the learned representations of protein language models as inputs.

The use of learned representations from protein language models in conjunction with advanced neural networks has also been explored. For example, PFresGO uses transformer layers to integrate protein features generated by one-hot encoding and ProtT5, a protein language model with 562 million parameters, with GO hierarchy features via a multi-head cross-attention mechanism [45,62]. GNNs, which update node representations by exchanging information with neighboring nodes, have also been used to extract more complex protein features from graph data, such as protein–protein interaction networks. Given that determining the role of a protein in BPs and its CC is challenging solely from its amino acid sequence, DeepGOGAT-SE uses a graph attention network to extract protein–protein interaction features [63]. By leveraging protein representations calculated from ESM2, a protein language model with 3 billion parameters, the graph attention network learns protein functions within the context of protein–protein interaction networks.

Advances in deep learning now enable the direct use of 3D protein structures, as described earlier. For instance, DeepFRI was developed to predict GO terms using both amino acid sequences and protein structures [64]. DeepFRI comprises two neural networks: (i) a long short-term memory network pretrained on amino acid sequences to extract residue-level features; and (ii) a GCN that propagates the residue-level features through a graph where nodes represent residues and edges connect adjacent residues (Figure 3D). The combination of sequence and structure information not only improved prediction performance, but also provided the interpretation of the deep learning model, identifying residues associated with predicted functions. When compared with existing GO term prediction tools (i.e., BLAST, DeepGO, FunFams, DeepFRI, TALE+, and DeepGOZero), PFresGO showed superior performance on BP and CC domains, achieving F_{max} scores of 0.5678 and 0.6737, respectively, on an independent test dataset. For the MF domain, PFresGO achieved a F_{max} score of 0.6917, which is similar to the highest score, 0.7191, achieved by DeepGOZero.

With the great progress in protein structure prediction, such as the development of AlphaFold and RoseTTAFold, numerous predicted protein structures have started to be used in GO term prediction models, enhancing prediction performance derived from big data [56,65]. For example, TransFun uses protein structures predicted by AlphaFold2 to extract adjacency matrices of amino acid residues, which are subsequently used to process residue-level representations from protein language models [66]. TransFun outperformed existing GO term prediction tools, such as DiamondScore, DeepGO, DeepGOCNN, TALE, and DeepFRI, on all GO domains, obtaining F_{max} scores of 0.659, 0.551, and 0.395 for CC, MF, and BP domains, respectively, while the other tools achieved F_{max} scores from 0.502 to 0.654, from 0.392 to 0.548, and from 0.362 to 0.398, respectively on the CAFA3 test dataset used as an independent dataset [67].

As the availability of diverse bio big data increases, AI for enzyme function classification has begun to incorporate various types of biological data beyond protein sequences alone. In a recent study, an LLM was pretrained on protein-coding DNA sequences instead of amino acid sequences, using masked language modeling to predict masked codons [68]. This approach demonstrated performance comparable or superior to that of protein language models in predicting protein features, such as solubility, subcellular localization, and GO term predictions, by leveraging latent features extracted from patterns of synonymous codon usage, which are not accessible from amino acid sequences. Likewise, the availability of extensive biological data and advances in deep learning have significantly enhanced the prediction of GO terms, shedding light on previously unknown aspects of protein function (Figure 3E).

Concluding remarks

Here, we reviewed the progress of AI in predicting enzyme functions. We began by examining conventional ML algorithms, such as kNN, SVM, and random forests, along with machine-readable representations of protein features commonly used as inputs. These features range from amino acid compositions and physicochemical properties to sequence profile-derived embeddings. By understanding these conventional algorithms, we explored how data-driven approaches have been used in functional genomics. Finally, we reviewed the development of deep learning models for prediction of enzyme functions and highlighted state-of-the-art neural network architectures and algorithms, including transformer-based protein language models.

Although deep learning has facilitated high-performance enzyme function prediction, model generalizability remains constrained by inherent imbalances in enzyme data. Data scarcity within certain enzyme classes limits the ability of deep learning models to effectively learn and generalize characteristic features. In addition, taxonomic bias, stemming from the over- or under-representation of specific taxa, further undermines generalizability by skewing predictions toward well-represented groups and reducing accuracy for under-represented or novel taxa. To address these challenges, data augmentation strategies can mitigate data scarcity by generating synthetic enzyme sequences through computational methods, such as generative adversarial networks and variational autoencoders, thereby enhancing class diversity. Furthermore, incorporating phylogenetic relationships into model training can reduce taxonomic bias. One approach involves phylogenetic tree-based regularization, ensuring that evolutionarily related enzymes adopt more coherent functional representations within the model.

Another challenge lies in the lack of interpretability in deep learning models. Most deep learning models operate as black-box systems, making it difficult to discern which specific features are pivotal in transforming raw data into predictive outputs. Consequently, these models generate predictions without offering interpretable insights into the underlying relationships between enzyme characteristics and their functions. Analyzing attention maps or feature importance evaluation methodologies, such as Shapley additive explanations, can enhance model interpretability and address these limitations. For instance, analyzing attention scores from the self-attention layer in a transformer-based model revealed that amino acid residues related to active sites and cofactor binding sites significantly contributed to EC number prediction [39].

Finally, although high-performance enzyme function prediction has been enabled based on well-established nomenclature systems (e.g., EC number and GO term), these frameworks limit the discovery of novel enzymatic functions. Given these advances, future directions of AI-based functional genomics should explore approaches that move beyond traditional nomenclature systems (see Outstanding questions). One possibility is a bottom-up strategy, which predicts enzyme functions without being constrained by predefined terms, thereby expanding the scope for discovering entirely new functionalities. In parallel, the rise of generative AI offers a powerful opportunity for designing novel enzymes with desired functions. By leveraging deep learning and bio big data, it can create *de novo* enzymes from scratch, bridging the gap between function prediction and applications of biotechnologies (Box 2). Together, these innovations will reshape the landscapes of functional genomics, enzymology, metabolic engineering, and synthetic biology, enabling numerous biotechnology applications in the fields of medicine, food, cosmetics, chemicals, materials, and other industrial biotechnologies [69–71].

Declaration of generative AI and AI-assisted technologies in the writing process

After writing the manuscript, ChatGPT was used solely to check English grammar. Following this grammar check, the authors reviewed and revised the content once more to ensure its correctness. The authors take full responsibility for the content of the published article.

Outstanding questions

What novel data representations or feature extraction methods offer the greatest potential for improving AI models in enzyme function prediction beyond traditional sequence-based embeddings?

What novel enzyme function nomenclature systems can be devised to capture the complexity of enzyme functions in AI-driven functional genomics?

How can AI be advanced from merely classifying enzyme functions to accurately predicting key functional parameters that inform the rational design of enzymes with desired properties?

What strategies can be implemented to integrate generative AI with experimental validation, ensuring that *de novo* enzyme designs based on AI predictions meet practical biochemical functionality, stability, and other desired characteristics?

Box 2. *De novo* enzyme design and following ethical issues

The rise of generative AI offers a powerful opportunity for designing novel enzymes with desired functions beyond the functional annotation of unknown sequences. For example, ProteinMPNN has enabled the precise extraction of amino acid sequences from protein structures using message-passing neural networks [88]. Furthermore, RFdiffusion and RFAA have demonstrated the ability to generate protein structures with binding capabilities for specific molecules by using denoising diffusion probabilistic models [89,90]. Building on these advances, generative AIs now facilitate the design of protein structures with desired functions and the identification of corresponding amino acid sequences required for their formation. However, enzyme design poses greater challenges than designing ligand-binding proteins, as it requires optimizing both substrate binding and catalytic activity to ensure functionality.

To address this challenge, research is focusing on integrating deep learning-based scaffold generation with catalytic site optimization to ensure the functionality of designed enzymes [91]. This approach has facilitated the development of artificial enzymes, such as *de novo* luciferase and serine hydrolase [92,93]. These approaches provide a systematic framework for engineering enzymes with enhanced activity and specificity, thereby expanding the potential of computational enzyme design in synthetic biology and industrial biotechnology. Looking forward, future research should aim to establish a generalized platform for enzyme design that enables precise engineering of catalytic functions with broad applicability across diverse biochemical reactions.

While the advancement of deep learning in enzyme design can open new frontiers in biotechnology, it also raises significant ethical concerns that must be addressed. One major concern is the potential ecological impact of artificially designed enzymes if they are unintentionally released (especially in the cell or organism) into natural environments. Unlike naturally evolved enzymes, computationally designed enzymes and the metabolisms they might alter may exhibit unforeseen interactions with ecosystems, potentially disrupting microbial communities or biochemical cycles in unexpected ways. In addition, the potential misuse of artificially designed enzymes can pose serious biosecurity risks. The ability to design enzymes with precise catalytic functions could be exploited to create biological weapons, illicit drugs, or novel psychoactive substances. This necessitates strict ethical considerations, regulatory oversight, and international collaboration to prevent misuse, while ensuring that these powerful tools are harnessed responsibly for scientific and industrial advances. As computational enzyme design continues to improve, maintaining a balance between innovation, biosecurity, and environmental responsibility is crucial for mitigating unintended consequences.

Acknowledgments

This work was supported by the Development of advanced synthetic biology source technologies for leading the biomanufacturing industry project (RS-2024-00399424), funded by the National Research Foundation and supported by the Korean Ministry of Science and ICT. Additionally, this work was supported by the Korea-US Collaborative Research Fund (KUCRF), which is funded by both the Korean Ministry of Science and ICT and the Korean Ministry of Health & Welfare (grant number: RS-2024-00468410).

Declaration of interests

The authors declare no competing interests.

Resources

[i]https://alphafold.ebi.ac.uk/

[ii]https://geneontology.org/stats.html

References

1. Altschul, S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.* 215, 403–410
2. Eddy, S.R. (1998) Profile hidden Markov models. *Bioinformatics* 14, 755–763
3. LeCun, Y. *et al.* (2015) Deep learning. *Nature* 521, 436–444
4. Zou, J. *et al.* (2019) A primer on deep learning in genomics. *Nat. Genet.* 51, 12–18
5. Glasner, M.E. *et al.* (2006) Evolution of enzyme superfamilies. *Curr. Opin. Chem. Biol.* 10, 492–497
6. Jordan, M.I. and Mitchell, T.M. (2015) Machine learning: trends, perspectives, and prospects. *Science* 349, 255–260
7. Jukes, T.H. *et al.* (1975) Amino acid composition of proteins: selection against the genetic code. *Science* 189, 50–51
8. Nasibov, E. and Kandemir-Cavas, C. (2009) Efficiency analysis of KNN and minimum distance-based classifiers in enzyme family prediction. *Comput. Biol. Chem.* 33, 461–464
9. Pradhan, D. *et al.* (2017) Enzyme classification using multiclass support vector machine and feature subset selection. *Comput. Biol. Chem.* 70, 211–219
10. Rice, P. *et al.* (2000) EMBOSS: the European molecular biology open software suite. *Trends Genet.* 16, 276–277
11. Li, Z.R. *et al.* (2006) PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence. *Nucleic Acids Res.* 34, W32–W37
12. Wilkins, M.R. *et al.* (1999) Protein identification and analysis tools in the ExPASy server. *Methods Mol. Biol.* 112, 531–552
13. Dalkiran, A. *et al.* (2018) ECPred: a tool for the prediction of the enzymatic functions of protein sequences based on the EC nomenclature. *BMC Bioinformatics* 19, 334
14. Cai, C.Z. *et al.* (2004) Enzyme family classification by support vector machines. *Proteins* 55, 66–76
15. Frishman, D. and Argos, P. (1995) Knowledge-based protein secondary structure assignment. *Proteins* 23, 566–579

16. McGuffin, L.J. *et al.* (2000) The PSIPRED protein structure prediction server. *Bioinformatics* 16, 404–405
17. Dobson, P.D. and Doig, A.J. (2005) Predicting enzyme class from protein structure without alignments. *J. Mol. Biol.* 345, 187–199
18. Concu, R. *et al.* (2009) Prediction of enzyme classes from 3D structure: a general model and examples of experimental-theoretic scoring of peptide mass fingerprints of *Leishmania* proteins. *J. Proteome Res.* 8, 4372–4382
19. Shen, H.B. and Chou, K.C. (2007) EzyPred: a top-down approach for predicting enzyme functional classes and subclasses. *Biochem. Biophys. Res. Commun.* 364, 53–59
20. Cover, T. and Hart, P. (1967) Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* 13, 21–27
21. De Ferrari, L. *et al.* (2012) EnzML: multi-label prediction of enzyme classes using InterPro signatures. *BMC Bioinformatics* 13, 61
22. Cortes, C. and Vapnik, V. (1995) Support-vector networks. *Mach. Learn.* 20, 273–297
23. Cai, C.Z. *et al.* (2003) SVM-Prot: web-based support vector machine software for functional classification of a protein from its primary sequence. *Nucleic Acids Res.* 31, 3692–3697
24. Quinlan, J.R. (1986) Induction of decision trees. *Mach. Learn.* 1, 81–106
25. Arakaki, A.K. *et al.* (2009) EFICAz^2: enzyme function inference by a combined approach enhanced by machine learning. *BMC Bioinformatics* 10, 107
26. Breiman, L. (2001) Random forests. *Mach. Learn.* 45, 5–32
27. Chen, L. *et al.* (2024) PredictEFC: a fast and efficient multi-label classifier for predicting enzyme family classes. *BMC Bioinformatics* 25, 50
28. Rosenblatt, F. (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.* 65, 386–408
29. Jensen, L.J. *et al.* (2002) Prediction of human protein function from post-translational modifications and localization features. *J. Mol. Biol.* 319, 1257–1265
30. Martinez Cuesta, S. *et al.* (2015) The classification and evolution of enzyme function. *Biophys. J.* 109, 1082–1086
31. Jabeen, S. *et al.* (2023) A review on methods and applications in multimodal deep learning. *ACM Trans. Multimed. Comput. Commun. Appl.* 19, 1–41
32. Li, Y. *et al.* (2018) DEEPre: sequence-based enzyme EC number prediction by deep learning. *Bioinformatics* 34, 760–769
33. Zou, Z. *et al.* (2018) mlDEEPre: multi-functional enzyme function prediction with hierarchical multi-label deep learning. *Front. Genet.* 9, 714
34. Gao, R. *et al.* (2019) Prediction of enzyme function based on three parallel deep CNN and amino acid mutation. *Int. J. Mol. Sci.* 20, 2845
35. Ryu, J.Y. *et al.* (2019) Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. *Proc. Natl. Acad. Sci. U. S. A.* 116, 13996–14001
36. Memon, S.A. *et al.* (2020) HECNet: a hierarchical approach to enzyme function classification using a Siamese Triplet Network. *Bioinformatics* 36, 4583–4589
37. Sanderson, T. *et al.* (2023) ProteInfer, deep neural networks for protein functional inference. *Elife* 12, e80942
38. Vaswani, A. *et al.* (2017) Attention is all you need. *Adv. Neural Inf. Proces. Syst.* 30, 6000–6010
39. Kim, G.B. *et al.* (2023) Functional annotation of enzyme-encoding genes using deep learning with transformer layers. *Nat. Commun.* 14, 7370
40. Buchfink, B. *et al.* (2021) Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat. Methods* 18, 366–368
41. UniProt Consortium (2023) UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Res.* 51, D523–D531
42. Devlin, J. (2018) Bert: pre-training of deep bidirectional transformers for language understanding. *arXiv*, Published online October 11, 2018. http://dx.doi.org/10.48550/arXiv.1810.04805
43. Rives, A. *et al.* (2021) Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. U. S. A.* 118, e2016239118
44. Brandes, N. *et al.* (2022) ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics* 38, 2102–2110
45. Elnaggar, A. *et al.* (2022) ProtTrans: toward understanding the language of life through self-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 44, 7112–7127
46. Buton, N. *et al.* (2023) Predicting enzymatic function of protein sequences with attention. *Bioinformatics* 39, btad620
47. Ferruz, N. *et al.* (2022) ProtGPT2 is a deep unsupervised language model for protein design. *Nat. Commun.* 13, 4348
48. Weissenow, K. *et al.* (2022) Protein language-model embeddings for fast, accurate, and alignment-free protein structure prediction. *Structure* 30, 1169–1177
49. Zhang, Z. *et al.* (2023) Protein representation learning by geometric structure pretraining. *arXiv*, Published online January 27, 2023. http://dx.doi.org/10.48550/arXiv.1810.04805
50. Lin, Z. *et al.* (2023) Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* 379, 1123–1130
51. Mansoor, S. *et al.* (2023) Zero-shot mutation effect prediction on protein stability and function using RoseTTAFold. *Protein Sci.* 32, e4780
52. Zhang, Z. *et al.* (2023) A systematic study of joint representation learning on protein sequences and structures. *arXiv*, Published online March 10, 2025. http://dx.doi.org/10.48550/arXiv.2303.06275
53. Hayes, T. *et al.* (2025) Simulating 500 million years of evolution with a language model. *Science* 387, 850–858
54. Jang, Y.J. *et al.* (2024) Accurate prediction of protein function using statistics-informed graph networks. *Nat. Commun.* 15, 6601
55. Yu, T. *et al.* (2023) Enzyme function prediction using contrastive learning. *Science* 379, 1358–1363
56. Baek, M. *et al.* (2021) Accurate prediction of protein structures and interactions using a three-track neural network. *Science* 373, 871–876
57. Jumper, J. *et al.* (2021) Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 583–589
58. Varadi, M. *et al.* (2022) AlphaFold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res.* 50, D439–D444
59. Song, Y. *et al.* (2024) Accurately predicting enzyme functions through geometric graph learning on ESMFold-predicted structures. *Nat. Commun.* 15, 8180
60. Price, M.N. *et al.* (2018) Mutant phenotypes for thousands of bacterial genes of unknown function. *Nature* 557, 503–509
61. Strodthoff, N. *et al.* (2020) UDSMProt: universal deep sequence models for protein classification. *Bioinformatics* 36, 2401–2409
62. Pan, T. *et al.* (2023) PFresGO: an attention mechanism-based deep-learning approach for protein annotation by integrating Gene Ontology inter-relationships. *Bioinformatics* 39, btad094
63. Kulmanov, M. *et al.* (2024) Protein function prediction as approximate semantic entailment. *Nat. Mach. Intell.* 6, 220–228
64. Gligorijevic, V. *et al.* (2021) Structure-based protein function prediction using graph convolutional networks. *Nat. Commun.* 12, 3168
65. Abramson, J. *et al.* (2024) Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* 630, 493–500
66. Boadu, F. *et al.* (2023) Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function. *Bioinformatics* 39, i318–i325
67. Zhou, N. *et al.* (2019) The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.* 20, 244
68. Outeiral, C. and Deane, C.M. (2024) Codon language embeddings provide strong signals for use in protein engineering. *Nat. Mach. Intell.* 6, 170–179
69. Lee, S.Y. *et al.* (2019) A comprehensive metabolic map for production of bio-based chemicals. *Nat. Catal.* 2, 18–33
70. Jang, W.D. *et al.* (2023) An interactive metabolic map of bio-based chemicals. *Trends Biotechnol.* 41, 10–14
71. Kim, G.B. *et al.* (2023) Metabolic engineering for sustainability and health. *Trends Biotechnol.* 41, 425–451
72. Shi, Z. *et al.* (2023) Enzyme commission number prediction and benchmarking with hierarchical dual-core multitask learning framework. *Research* 6, 0153

73. Rong, D. *et al.* (2024) Autoregressive enzyme function prediction with multi-scale multi-modality fusion. *arXiv*, Published online August 11, 2024. http://dx.doi.org/10.48550/arXiv.2408.06391
74. Merino, G.A. *et al.* (2022) Hierarchical deep learning for predicting GO annotations by integrating protein knowledge. *Bioinformatics* 38, 4488–4496
75. Kulmanov, M. and Hoehndorf, R. (2022) DeepGOZero: improving protein function prediction from sequence and zero-shot learning based on ontology axioms. *Bioinformatics* 38, i238–i245
76. You, R. *et al.* (2021) DeepGraphGO: graph neural network for large-scale, multispecies protein function prediction. *Bioinformatics* 37, i262–i271
77. Wu, Z. *et al.* (2023) CFAGO: cross-fusion of network and attributes based on attention mechanism for protein function prediction. *Bioinformatics* 39, btad123
78. Liu, J. *et al.* (2023) Grain protein function prediction based on self-attention mechanism and bidirectional LSTM. *Brief. Bioinform.* 24, bbac493
79. Kulmanov, M. and Hoehndorf, R. (2020) DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics* 36, 422–429
80. Zhou, G. *et al.* (2020) Predicting functions of maize proteins using graph convolutional network. *BMC Bioinformatics* 21, 420
81. Yao, S. *et al.* (2021) NetGO 2.0: improving large-scale protein function prediction with massive sequence, text, domain, family and network information. *Nucleic Acids Res.* 49, W469–W475
82. Wang, S. *et al.* (2023) NetGO 3.0: protein language model improves large-scale functional annotations. *Genomics Proteomics Bioinformatics* 21, 349–358
83. Zhang, Z. *et al.* (2024) Structure-informed protein language model. *arXiv*, Publsihed online February 7, 2024. http://dx.doi.org/10.48550/arXiv.2402.05856
84. Zhang, X. *et al.* (2023) HNetGO: protein function prediction via heterogeneous network transformer. *Brief. Bioinform.* 24, bbab556
85. Kulmanov, M. *et al.* (2018) DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics* 34, 660–668
86. Lai, B. and Xu, J. (2022) Accurate protein function prediction via graph attention networks with predicted structure information. *Brief. Bioinform.* 23, bbab502
87. Zheng, R. *et al.* (2023) Large-scale predicting protein functions through heterogeneous feature fusion. *Brief. Bioinform.* 24, bbad243
88. Dauparas, J. *et al.* (2022) Robust deep learning-based protein sequence design using ProteinMPNN. *Science* 378, 49–56
89. Watson, J.L. *et al.* (2023) *De novo* design of protein structure and function with RFdiffusion. *Nature* 620, 1089–1100
90. Krishna, R. *et al.* (2024) Generalized biomolecular modeling and design with RoseTTAFold All-Atom. *Science* 384, eadl2528
91. Notin, P. *et al.* (2024) Machine learning for functional protein design. *Nat. Biotechnol.* 42, 216–228
92. Yeh, A.H. *et al.* (2023) *De novo* design of luciferases using deep learning. *Nature* 614, 774–780
93. Lauko, A. *et al.* (2025) Computational design of serine hydrolases. *Science*, eadu2454