Essential Genes Identification Model Based on Sequence Feature Map and Graph Convolutional Neural Network

Wenxing Hu
Gannan Normal University
Haiyang Xiao
Gannan Normal University
Lixin Guan
Gannan Normal University
Mengshan Li (msli@gnnu.edu.cn)
Gannan Normal University

Research Article

Keywords: Essential genes, Graphical convolutional neural networks, Machine learning, Gene sequences, Bioinformatics

Posted Date: July 3rd, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3077142/v1

Abstract

Background

Essential genes encode functions that play a vital role in the life activities of organisms, encompassing growth, development, immune system functioning, and cell structure maintenance. Conventional experimental techniques for identifying essential genes are resource-intensive and time-consuming, and the accuracy of current machine learning models needs further enhancement. Therefore, it is crucial to develop a robust computational model to accurately predict essential genes.

Results

In this study, we introduce GCNN-SFM, a computational model for identifying essential genes in organisms, based on graph convolutional neural networks (GCNN). GCNN-SFM integrates a graph convolutional layer, a convolutional layer, and a fully connected layer to model and extract features from gene sequences of essential genes. Initially, the gene sequence is transformed into a feature map using coding techniques. Subsequently, a multi-layer GCN is employed to perform graph convolution operations, effectively capturing both local and global features of the gene sequence. Further feature extraction is performed, followed by integrating convolution and fully-connected layers to generate prediction results for essential genes. The gradient descent algorithm is utilized to iteratively update the cross-entropy loss function, thereby enhancing the accuracy of the prediction results. Meanwhile, model parameters are tuned to determine the optimal parameter combination that yields the best prediction performance during training.

Conclusions

Experimental evaluation demonstrates that GCNN-SFM surpasses various advanced essential gene prediction models and achieves an average accuracy of 94.53%. This study presents a novel and effective approach for identifying essential genes, which has significant implications for biology and genomics research.

1 Introduction

Essential genes, which are currently a hot topic in genomics and bioinformatics research, are indispensable for supporting cellular life(1). Their coding functions are crucial for the survival of organisms. These genes constitute a set that must be present in an organism and are vital for maintaining its life activities under specific environmental conditions. They encode key proteins or RNA molecules that are essential for life, and their functions are considered fundamental for the organism's survival(2). In humans and other organisms, the functions of essential genes are often associated with basic cellular metabolism, growth, development, the immune system, and the maintenance of cellular

structure. Therefore, the study of essential genes is of great importance for our understanding of the fundamental physiological functions of organisms and the mechanisms of disease occurrence(3, 4).

With the completion of whole-genome sequencing and the development of genome-scale gene inactivation techniques, it has become possible to identify essential genes within the genome. Traditional experimental techniques used to identify essential genes(5) in organisms include gene knockout(6, 7) and gene silencing(8). Gene knockout is the process of inactivating a specific gene in an organism to observe its effects on the organism's survival and function. This can be accomplished through various techniques, including CRISPR-Cas9 gene editing(9). The aim of knockout is to determine whether a gene is an essential gene, that is, whether the absence of the gene would make the organism non-viable. On the other hand, gene silencing is used to study the function of a gene by interfering with or suppressing its expression, often accomplished through methods such as RNA interference(10) and antibodies(11). However, these traditional experimental methods still have several potential drawbacks: they are expensive, time-consuming, and do not offer comprehensive genome coverage. In modern biological research, machine learning models have been developed to computationally identify essential genes(12). These methods have been extensively employed to study essential genes and contribute to advancing our understanding of gene function and organismal complexity.

In machine learning methods for predicting essential genes, feature extraction is a key step that involves extracting useful feature information from genomic data for model learning. This feature information is combined with machine learning classification algorithms (SVM(13), NB(5, 13–15), RF(13, 16), etc.) to build models for essential gene prediction. High-throughput genome sequencing and homology localization(17) provide a variety of biological features for predicting essential genes, including network topology information(18, 19), homology information(20, 21), gene expression information(22, 23), and functional domains(23). For instance, Deng et al. developed an integrated classifier for essential genes by integrating information from diverse features extracted from different aspects of the essential genome sequence(24). Chen and Xu also successfully combined high throughput data with machine learning methods to determine protein deficiencies in Saccharomyces cerevisiae(25). Seringhaus et al. used various intrinsic genomic features to train machine learning models to predict essential genes in brewer's yeast(26), and Yuan et al. developed three machine learning methods to predict lethality in mouse knockouts based on informative genomic features, among others(27). However, these data are often not available(28, 29), and some data features do not have high predictive power or even add biological redundancy. Consequently, most models are currently constructed based on DNA sequence features of essential genes(30). For instance, Ning et al. employed single nucleotide frequencies, dinucleotide frequencies, and amino acid frequencies of gene sequences to predict essential genes in bacteria(31). Guo et al. emphasized the significance of local nucleotide composition and internal nucleotide association, proposing an approach known as λ-interval Z-curve to integrate both types of information(32). Chen et al. combined Z-curve pseudo-k-tuple nucleotide composition with an SVM classifier to construct a model aimed at capturing DNA sequence patterns associated with essential genes(33). Moreover, various other approaches have been explored, such as using natural language processing methods and integrating deep neural networks, as demonstrated by Le et al.(34) Overall, there

is a growing body of research utilizing machine learning methods for essential gene prediction(35, 36), which has led to significant improvements in prediction performance. However, further enhancements are still necessary, and new research methods are required to elucidate the crucial impact of gene sequences on essential gene prediction.

While machine learning-based approaches successfully predict essential genes, they exhibit significant variations in terms of the methods used for sequence feature extraction and the employed model structures. The predictive performance of a method relies on its ability to explore gene feature information and integrate it into the model structure effectively. Thus, enhancing model performance is critical in investigating novel methods. This paper introduces GCNN-SFM, a graph convolutional neural network-based method for identifying essential genes. GCNN-SFM converts gene sequence coding into a sequence feature map. It utilizes a combination of graph convolutional, convolutional, and fully connected layers to effectively capture and learn both local and global features in sequences, enabling accurate identification of essential genes. To ensure a robust model, we tuned its parameters and selected the combination that resulted in the best predictive performance during training. Additionally, we iteratively optimized the cross-entropy loss function using a gradient descent algorithm, thereby enhancing the identification of essential genes in organisms.

2 Theory and computational section

2.1 Datasets

In bioinformatics research, generalized benchmark datasets are crucial for constructing high-performance predictive models. In this study, we utilized datasets from four species: *Drosophila melanogaster* (D.melanogaster),*Methanococcus maripaludis* (M.maripaludis), *Caenorhabditis elegans*(C.elegans(37)), and *Homo sapiens* (H.sapiens). These datasets represent highly comprehensive resources in this specific field. Campos et al. curated comprehensive genomic data and associated annotations for D.melanogaster from sources such as FlyBase(http://ftp.flybase.net/genomes/Drosophila_melanogaster/)(38),Ensembl databases (https://ftp.ensembl.org/pub/current_fasta/drosophila_melanogaster/)(39), and peer-reviewed journal articles(40). Similarly, data for C.elegans were collected from WormBase(https://wormbase.org/species/c_elegans#1402--10)(41),Ensembl databases (https://ftp.ensembl.org/pub/current_fasta/caenorhabditis_elegans/), and peer-reviewed journal articles(42). Chen et al.(33) obtained the complete genome of M.maripaludis from the DEG (Database of Essential Genes: https://tubic.org/deg/public/index.php)(43), a comprehensive repository encompassing all available essential gene information. To reduce data redundancy and mitigate homology bias, sequences exhibiting over 80% structural similarity were excluded from the DEG. Furthermore, gene data for H. sapiens were extracted from the DEG database by Guo et al.(32) The dataset defined by each of you above is therefore chosen for this paper, and the baseline dataset is represented as follows:

$$\mathbb{S} = \mathbb{S}^{+} \cup \mathbb{S}^{-}$$

1

The data set for a species, denoted as $\mathbb{S}$, consists of two subsets: $\mathbb{S}^{+}$represents the positive data subset for essential genes, and $\mathbb{S}^{-}$represents the negative data subset for non-essential genes. The concatenation of these two subsets is defined as $\cup$.The provided dataset was divided into three sets: a training set, a validation set, and a test set, with a ratio of 8:1:1. The validation set assesses the model's generalization ability and detects overfitting during training, while the test set evaluates the model's performance after the completion of training. The details of the datasets are presented in Table.1.

Table.1 Number of gene sequences in the datasets

Dataset	Train set	Verification set	Test set	Reference
D. melanogaster	5425	829	1044	(40)
H.sapiens	9575	1064	1336	(32)
M. maripaludis	1271	141	178	(33)
C.elegans	12292	2476	3694	(42)

2.2 *Gapped k-mer* encoding feature extraction

The model predicts essential genes by encoding the gene sequence into the matrix format required for deep learning. Features are extracted from the gene sequence of an essential gene using *Gapped k-mers*(44, 45) encoding, resulting in a graph structure. In the field of bioinformatics, *k-mers* refer to subsequences of length k that are found within biological sequences. For example, in Table.2, a sequence with a length of L can be divided into *L-k + 1 k-mers*, depending on the value of *k*.

Table.2k -mer for gene sequence

Gene sequence: GTACTA	
k	*k-mer*
1	G,T,A,C,T,A
2	GT,TA,AC,CT,TA
3	GTA,TAC,ACT,CTA
4	CTAC,TACT,ACTA
5	GTACT,TACTA
6	GATCTA

To address the genetic variation that often occurs in biological sequences, in this study, we specifically examine bases that are separated by a distance of d unrelated positions within the sequences. Referring

to Table.2, the subsequence GTA can be represented as GT**A when *k = 3* and *d = 3*, with *representing the allowable distance within the gene sequence. The frequency of occurrence of all *k-mers* in this sequence was extracted, and a graph vector was constructed as the input. Initially, the gene sequence is divided into groups of bases based on the selected *k-mer* length. The number of occurrences of the k-valued base groups and the frequency of concatenated adjacent base groups is calculated. Finally, the subsequences of length k are combined in the order of the bases within the gene sequence, forming a graph structure. Eq. (1) is used to depict the structure of the graph.

$$G = (n, e)\,(1)$$

Here, $n$represents the set of nodes, while$e$represents the set of connected edges. The characteristic information of each node is determined by the frequency of occurrence of its respective *k-mer*, whereas the characteristic information of a connected edge is determined by the frequency of occurrence of two *k-mers* together. When *k = 3*, as illustrated in Fig. 1, the gene sequence of the essential gene can be transformed into a graph structure.

2.3 Models based on sequence feature maps and graph convolutional neural networks

In this study, we utilize the Graph Convolutional Neural Network (GCNN) as a model for predicting essential genes, referred to as GCNN-SFM. After applying the above encoding scheme, the gene sequences are transformed into graph structures. Graph Convolutional Neural Network is a deep learning model capable of processing graph data to perform feature learning and prediction tasks. Unlike traditional convolutional neural networks (CNNs), graph convolutional neural networks can handle irregular graph data with arbitrary connectivity relationships. The graph convolution layer employs graph convolution operations to aggregate information from neighboring nodes and update the node representation. This convolution layer can extract features unaffected by coding space transformations. Additionally, the fully connected layer processes the information extracted by the upstream convolution layer in a non-linear manner. The structure of GCNN-SFM is depicted in Fig. 2.

The GCNN-SFM model adopts a progressive learning approach to capture the representation of nodes across multiple layers of graph convolution. Each layer of the graph convolution comprises two steps: neighbor aggregation and feature transformation. In the neighbor aggregation step, the feature information from each node v_i is aggregated from its neighboring nodes. The aggregation operation involves weighted summation, where the weights are determined based on the features of the neighboring nodes and the edge weights. The specific formula is presented in Eq. (2).

$$Z_i^{(k)} = \sum\nolimits_{j \in N(i)} \frac{1}{\sqrt{d_i d_j}} \bullet h_j^{(k-1)} \bullet W^{(k-1)}\,(2)$$

The aggregated features of nodev_i at layer k, denoted as $Z_i^{(k)}$, are derived. $N\left(i\right)$represents the set of neighboring nodes connected to nodev_i. The degrees of nodes v_i and $v_j$are respectively represented by $d_i$and d_j. Furthermore, $h_j^{(k-1)}$ refers to the features of nodev_j at the previous layer, k-1. Additionally, $W^{(k-1)}$denotes the weight matrix used for the linear transformation of these features.

In the feature transformation, we perform a linear transformation on the aggregated features and apply a non linear activation function to introduce non linear expressivity. The feature transformation can be expressed as Eq. (3):

$$H_i^{(k)} = \sigma\left(Z_i^{(k)}\right)$$

3

Where $H_i^{(k)}$ represents the representation matrix of node v_i at layer k, and σ denotes the nonlinear activation function, specifically the rectified linear unit (ReLU). By applying weighted aggregation and nonlinear transformation to the neighboring nodes, the new feature representation $H_i^{(k)}$ of the current layer's node can be obtained. The GCNN-SFM employs multi-layer graph convolution operations to progressively aggregate and propagate information from the node's neighbors, enriching its feature representation. Subsequently, this representation undergoes convolution layer operations, reshaping it into a tensor x that aligns with the input shape. Finally, it is fed into a fully connected layer to be mapped to the label space of the prediction task, as demonstrated in Eq. (4).

$$\hat{y} = softmax\left(ReLU\left(W_1 \bullet x + b_1\right) \bullet W_2 + b_2\right)$$

4

Where $\hat{y}$ represents the predicted gene label by the model, W_1 and b_1 refer to the weight matrix and bias vector of the first fully connected layer. Similarly, $W_2$and $b_2$represent the weight matrix and bias vector of the second fully connected layer, respectively.

To establish this mapping, it is necessary to define a loss function that measures the discrepancy between the predicted labels and the true labels. This loss function is iteratively updated using gradient descent to minimize the loss and enhance the accuracy of the predictions made by the GCNN-SFM. In this study, the selected loss function is the widely employed cross-entropy loss, commonly used in multi-classification problems.

$$\mathcal{L} == -\frac{1}{N}\sum_{n=1}^{N}\left(y^{(n)}logp^{(n)} + (1-y^{(n)})log(1-p^{(n)})\right)$$

5

Where N is the sample size, $y^{(n)}$is the binary variable, and $p^{(n)}$ is the probability that the neural network predicts the nth sample as an essential gene.

2.4 Model Performance Evaluation

To evaluate the classification performance of the model, we employ several commonly used metrics, consistent with the approach taken by Le et al(34). These metrics encompass sensitivity (SN), specificity (SP), accuracy (ACC), Matthew correlation coefficient (MCC), and area under the receiver operating characteristic curve (AUC). The specific calculation procedures for each metric are outlined below.

$$\begin{cases} SN = \frac{TP}{TP+FN} \times 100\% \\ SP = \frac{TN}{TN+FP} \times 100\% \\ ACC = \frac{TP+TN}{TP+FN+TN+FP} \times 100\% \\ MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}} \\ AUC = \frac{\sum_{i \in pos} rank_i - \frac{num_{pos}(num_{pos}+1)}{2}}{num_{pos} num_{neg}} \end{cases}$$

6

Among them, TP, TN, FP and FN represent the number of samples whose prediction results are true positive, true negative, false positive and false negative, respectively. The AUC (Area Under Curve) is defined as the area under the ROC curve, enclosed by the coordinate axes. The closer the AUC value is to 1.0, the better the model's performance.

3 Results and discussion

3.1 Experimental results for different parameters of sequence coding

In most machine learning and deep learning tasks, the encoding method plays a crucial role in obtaining high-quality models. The parameters *k* and *d* in the sequence coding method determine the quality of the sequence feature map. To identify the optimal parameter combination, we conducted preliminary experiments on the data. We combined various values of *k* and *d*, applied 10-fold cross-validation to validate a dataset consisting of four species, and evaluated the trained models. The relevant information of the used dataset is shown in Table.1 and the results obtained are presented in Fig. 3.

Firstly, to determine the optimal parameter settings for the graph coding method and achieve accurate prediction of essential genes, we defined various parameter combinations (*k = 2, d = 2; k = 2, d = 3; k = 3, d = 2; k = 3, d = 3*) that were likely to yield optimal performance. Setting the parameters *k* and *d* too high can result in overfitting of the trained model. Figure 3(b) and Fig. 3(c) demonstrate that when both *k* and *d* are

set to 3, the model predicts higher values of specificity (SP) and accuracy (ACC) compared to other parameter combinations for essential genes across the four species. The sensitivity (SN) value for M. maripaludis species in Fig. 3(a) is significantly higher, reaching 90%, compared to the other three parameter combinations. These findings suggest that the graph coding method with parameters set to (*k = 3, d = 3*) enables more efficient learning of DNA sequence features for essential genes by the model. From Fig. 3(g) and Fig. 3(h), it is evident that the model integrated with the graph coding method using parameter (*k = 3, d = 3*) outperformed other parameter combinations, achieving the highest performance across all datasets, with an average accuracy (ACC) of 94.53% and an area under the curve (AUC) of 82.99%. These findings indicate that utilizing the graph coding method with parameters set to (*k = 3, d = 3*) enables a more accurate representation of gene sequence characteristics, resulting in superior predictive performance of the model.

3.2 Experimental results for different datasets

To assess the performance of our proposed model GCNN-SFM, we conducted experiments using independent datasets from four species (D.melanogaster, M.maripaludis, H.sapiens, C.elegans) to assess its stability. Based on the results of previous experiments, the model outperformed other parameter combinations when the graph coding method was set to (*k = 3, d = 3*). Hence, we selected (*k = 3, d = 3*) as the optimal parameter configuration for subsequent experiments. The models underwent training and validation through a 10-fold cross-validation process using the training dataset. Prior to this, the DNA gene sequences were transformed into feature matrices using coding methods to facilitate the training and validation of the deep learning models. The trained models were then tested and evaluated on independent test sets, and the predictive performance of each independent dataset is illustrated in Fig. 4.

The GCNN-SFM model exhibited excellent performance for various species, as shown in the experimental results depicted in Fig. 4. Notably, Fig. 4(c) illustrates that the ACC values for predicting essential genes using the model surpassed 90% for all four species, with the D.melanogaster species achieving an exceptionally high ACC value of 98.47%. This finding affirms the validity of the essential gene prediction model. Conversely, in the case of the C. elegans species, as observed in Fig. 4(d) and Fig. 4(e), lower MCC and AUC values were noted compared to those of other species, yet a maintained ACC value of 92.42% was observed. Upon analyzing the SN values, it is hypothesized that the marginally lower MCC and AUC values observed for the C.elegans species result from the limited availability of essential gene data specific to C.elegans. Overall, the model demonstrated remarkable performance across the four species, as illustrated in Fig. 4(f) and Table.3, attaining an average ACC value of 94.53%. These results underscore the stability and reliability of our method, validating its effectiveness as a powerful tool for essential gene prediction.

Table.3 Prediction of experimental results for different species and mean values

Dataset	SN	SP	ACC	MCC	AUC
D.melanogaster	0.8333	0.9939	0.9847	0.8545	0.8283
M.maripaludis 4mC_F.vesca	0.9052	0.9304	0.9221	0.8265	0.8422
H.sapiens	0.9048	0.9566	0.9501	0.7961	0.8655
C.elegans	0.8362	0.9368	0.9242	0.6983	0.7834
Average	0.8699	0.9544	0.9453	0.7939	0.8299

3.3 Results of cross-species validation experiments

To investigate whether the DNA sequences of essential genes exhibit specific characteristics or sequence similarities across species, we conducted cross-species validation experiments. This is shown in Fig. 5.

Using independent datasets from four species (D.melanogaster, M.maripaludis, H.sapiens, and C.elegans), we trained the DNA gene sequences of one species and evaluated the DNA gene sequences of another species to predict whether they were essential genes. The obtained results are depicted in Fig. 6, where the horizontal axis represents the training set, and the vertical axis represents the test set.

Figure 6(d) demonstrates the high accuracy (ACC) observed in two species: D.melanogaster and C.elegans. Training the model with a dataset from the species C.elegans and testing it with D.melanogaster resulted in a model prediction accuracy of 91.83% (ACC). Similarly, training the model with a dataset from D.melanogaster and testing it with C.elegans yielded predictions with an ACC value of 85.1%, suggesting a comparable pattern of nucleotide distribution between the two species. D.melanogaster, C.elegans, M.maripaludis, and H.sapiens exhibited low values for SN, ACC, and AUC, signifying substantial differences in nucleotide distribution among these species. These findings align with the genetic similarity results reported by Campos et al.(46), indicating striking similarities in nucleotide patterns among essential genes in certain species.

3.4 Experimental results comparing performance with other existing methods

To evaluate the effectiveness of our proposed model GCNN-SFM in identifying essential genes, we conducted a comparison with published models that address the same problem, including Pheg, iEsGene-ZCPseKNC, and eDNN-EG. Table.4 displays the pertinent information for each of the compared models.

Table.4 Information on each comparison model

Model	Description	Dataset	Reference
Pheg	Combining Z-curve and nucleotide composition learning features for k-intervals using SVM as a classifier	M.maripaludis	(32)
iEsGene-ZCPseKNC	Combining Z-curve and pseudo-k-tuple nucleotide composition learning features using SVM as a classifier	M.maripaludis	(33)
eDNN-EG	Natural language processing model learning features, integrating supervised learning models	M.maripaludis	(34)
GCNN-SFM	Gapped k-mer encodes sequences into graph features, combined with graph convolutional neural networks	M.maripaludis	-

As shown in Table.4, we performed experiments on the M.maripaludis dataset, and all predictor variables being compared were executed and evaluated on the independent M.maripaludis dataset. The results of the comparisons are shown in Fig. 7.

As shown in Fig. 7, GCNN-SFM outperforms eDNN-EG, iEsGene-ZCPseKNC, and Pheg models. Compared to these models, GCNN-SFM exhibits increased ACC values of 14.89%, 17.67%, and 17.09%, respectively. While the SN values of eDNN-EG, iEsGene-ZCPseKNC, and Pheg are significantly lower than their corresponding SP values, GCNN-SFM achieves a higher SN value of 90.52%. The SP value of GCNN-SFM does not differ significantly from that of the other models. The lower SN values of eDNN-EG, iEsGene-ZCPseKNC, and Pheg can be attributed to the considerable imbalance between the numbers of essential gene samples and non-essential gene samples in each training cycle. To address this imbalance, our GCNN-SFM model exclusively employs a sample class weighting strategy during the cross-validation process, preventing overfitting. Consequently, our model achieves an SN value that closely approximates the SP value during prediction. These results demonstrate that the GCNN-SFM model enhances the accuracy of predicting essential genes and outperforms other existing prediction methods.

4 Conclusions

This study proposes a graph convolutional neural network (GCNN)-based approach for essential gene prediction. The model GCNN-SFM effectively captures and learns local and global features in gene sequences through graph modeling and feature extraction, enabling the accurate identification of essential genes. The experimental results demonstrate significant performance advantages of our approach in tasks related to essential gene prediction. Our approach excels at extracting more discriminative feature representations in genes compared to traditional methods that rely on sequence feature engineering. Furthermore, this study unveils the potential of GCNN in predicting essential genes, thereby offering a new pathway for comprehending gene function and disease pathogenesis at a deeper level. There are some important considerations to address in future research. Firstly, the model may encounter computational challenges when dealing with large-scale genomic datasets, requiring further optimization and acceleration for practical applications. Secondly, the accuracy of the gene annotation

information of the GCNN-SFM model is crucial and has a significant impact on the prediction performance. Future research can be enhanced and expanded by integrating multimodal data sources, such as gene expression data and protein interaction networks(15, 47, 48), to enhance the prediction accuracy and robustness. In summary, this study offers robust support for further exploring gene regulatory networks and mechanisms of related diseases by enhancing our understanding of gene function and the prediction of essential genes.

Declarations

Ethics approval and consent to participate: Not applicable

Consent for publication: Not applicable

Availability of data and materials:

A static version of the package D.melanogaster and C.elegans datasets containing data linked to this publication is available at: (https://doi.org/10.6084/m9.figshare.12061815)and (https://doi.org/10.6084/m9.figshare.11533101).The codes, dataset, architecture, parameters, functions, usage and output of the proposed model are available free of charge at GitHub. (https://github.com/xing1999/GCNN-SFM).

Competing interests: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding: This research was funded by National Natural Science Foundation of China (Grant Numbers: 51663001, 52063002, 42061067 and 61741202).

Author contributions: Wenxing Hu and Mengshan Li designed the study; Wenxing Hu and Haiyang Xiao performed the research; Mengshan Li conceived the idea; Lixin Guan and provided and analyzed the data; Mengshan Li and Lixin Guan helped perform the analysis with constructive discussions; all authors contributed to writing and revision.

Acknowledgements: The authors gratefully acknowledge the support from the National Natural Science Foundation of China (Grant Numbers: 51663001, 52063002, 42061067 and 61741202) and we thank LetPub (www.letpub.com) for its linguistic assistance during the preparation of this manuscript.

References

1. O'Neill RS, Clark DV. The Drosophila melanogaster septin gene Sep2 has a redundant function with the retrogene Sep5 in imaginal cell proliferation but is essential for oogenesis. Genome. 2013;56(12):753–8.
2. Kobayashi K, Ehrlich SD, Albertini A, Amati G, Andersen K, Arnaud M et al. Essential Bacillus subtilis genes. Proceedings of the National Academy of Sciences. 2003;100(8):4678-83.

3. Juhas M, Eberl L, Glass JI. Essence of life: essential genes of minimal genomes. Trends Cell Biol. 2011;21(10):562–8.
4. Juhas M, Reuß DR, Zhu B, Commichau FM. Bacillus subtilis and Escherichia coli essential genes and minimal cell factories after one decade of genome engineering. Microbiology. 2014;160(11):2341–51.
5. Gustafson AM, Snitkin ES, Parker SC, DeLisi C, Kasif S. Towards the identification of essential genes using targeted genome sequencing and comparative analysis. BMC Genomics. 2006;7(1):1–16.
6. Giaever G, Chu AM, Ni L, Connelly C, Riles L, Véronneau S, et al. Functional profiling of the Saccharomyces cerevisiae genome. Nature. 2002;418(6896):387–91.
7. Roemer T, Jiang B, Davison J, Ketela T, Veillette K, Breton A, et al. Large-scale essential gene identification in Candida albicans and applications to antifungal drug discovery. Mol Microbiol. 2003;50(1):167–81.
8. Rancati G, Moffat J, Typas A, Pavelka N. Emerging and evolving concepts in gene essentiality. Nat Rev Genet. 2018;19(1):34–49.
9. Sidik SM, Huet D, Ganesan SM, Huynh M-H, Wang T, Nasamu AS, et al. A genome-wide CRISPR screen in Toxoplasma identifies essential apicomplexan genes. Cell. 2016;166(6):1423–35. e12.
10. Cullen LM, Arndt GM. Genome-wide screening for gene function using RNAi in mammalian cells. Immunol Cell Biol. 2005;83(3):217–23.
11. Friedel RH, Soriano P. Gene trap mutagenesis in the mouse. Methods in enzymology. 477: Elsevier; 2010. 243–69.
12. Mobegi FM, Zomer A, De Jonge MI, Van Hijum SA. Advances and perspectives in computational prediction of microbial gene essentiality. Brief Funct Genomics. 2017;16(2):70–9.
13. Lloyd JP, Seddon AE, Moghe GD, Simenc MC, Shiu S-H. Characteristics of plant essential genes allow for within-and between-species prediction of lethal mutant phenotypes. Plant Cell. 2015;27(8):2133–47.
14. Kim W. Prediction of essential proteins using topological properties in GO-pruned PPI network based on machine learning methods. Tsinghua Sci Technol. 2012;17(6):645–58.
15. Zhong J, Wang J, Peng W, Zhang Z, Pan Y. Prediction of essential proteins based on gene expression programming. BMC Genomics. 2013;14(4):1–8.
16. Nigatu D, Sobetzko P, Yousef M, Henkel W. Sequence-based information-theoretic features for gene essentiality prediction. BMC Bioinformatics. 2017;18(1):1–11.
17. Hua H-L, Zhang F-Z, Labena AA, Dong C, Jin Y-T, Guo F-B. An approach for predicting essential genes using multiple homology mapping and machine learning algorithms. BioMed research international. 2016;2016.
18. Acencio ML, Lemke N. Towards the prediction of essential genes by integration of network topology, cellular localization and biological process information. BMC Bioinformatics. 2009;10(1):1–18.

19. Plaimas K, Eils R, König R. Identifying essential genes in bacterial metabolic networks with machine learning methods. BMC Syst Biol. 2010;4(1):1–16.
20. Wei W, Ning L-W, Ye Y-N, Guo F-B. Geptop: a gene essentiality prediction tool for sequenced bacterial genomes based on orthology and phylogeny. PLoS ONE. 2013;8(8):e72343.
21. Song K, Tong T, Wu F. Predicting essential genes in prokaryotic genomes using a linear method: ZUPLS. Integr Biology. 2014;6(4):460–9.
22. Cheng J, Xu Z, Wu W, Zhao L, Li X, Liu Y, et al. Training set selection for the prediction of essential genes. PLoS ONE. 2014;9(1):e86805.
23. Deng J, Deng L, Su S, Zhang M, Lin X, Wei L, et al. Investigating the predictability of essential genes across distantly related organisms using an integrative approach. Nucleic Acids Res. 2011;39(3):795–807.
24. Deng J. An integrated machine-learning model to predict prokaryotic essential genes. Gene Essentiality: Methods and Protocols. 2015:137 – 51.
25. Chen Y, Xu D. Understanding protein dispensability through machine-learning analysis of high-throughput data. Bioinformatics. 2005;21(5):575–81.
26. Seringhaus M, Paccanaro A, Borneman A, Snyder M, Gerstein M. Predicting essential genes in fungal genomes. Genome Res. 2006;16(9):1126–35.
27. Yuan Y, Xu Y, Xu J, Ball RL, Liang H. Predicting the lethal phenotype of the knockout mouse by integrating comprehensive genomic data. Bioinformatics. 2012;28(9):1246–52.
28. Liao Q, Zhang Q. Local coordinate based graph-regularized NMF for image representation. Sig Process. 2016;124:103–14.
29. Su S, Zhang L, Liu J. An effective method to measure disease similarity using gene and phenotype associations. Front Genet. 2019;10:466.
30. Aromolaran O, Beder T, Oswald M, Oyelade J, Adebiyi E, Koenig R. Essential gene prediction in Drosophila melanogaster using machine learning approaches based on sequence and functional features. Comput Struct Biotechnol J. 2020;18:612–21.
31. Ning L, Lin H, Ding H, Huang J, Rao N, Guo F. Predicting bacterial essential genes using only sequence composition information. Genet Mol Res. 2014;13(2):4564–72.
32. Guo FB, Dong C, Hua HL, Liu S, Luo H, Zhang HW, et al. Accurate prediction of human essential genes using only nucleotide composition and association information. Bioinformatics. 2017;33(12):1758–64.
33. Chen J, Liu Y, Liao Q, Liu B. iEsGene-ZCPseKNC: Identify Essential Genes Based on Z Curve Pseudo k -Tuple Nucleotide Composition. IEEE Access. 2019;7:165241–7.
34. Le NQK, Do DT, Hung TNK, Lam LHT, Huynh TT, Nguyen NTK. A Computational Framework Based on Ensemble Deep Neural Networks for Essential Genes Identification. Int J Mol Sci. 2020;21(23).
35. Aromolaran O, Aromolaran D, Isewon I, Oyelade J. Machine learning approach to gene essentiality prediction: a review. Brief Bioinform. 2021;22(5):bbab128.

36. Campos TL, Korhonen PK, Hofmann A, Gasser RB, Young ND. Harnessing model organism genomics to underpin the machine learning-based prediction of essential genes in eukaryotes - Biotechnological implications. Biotechnol Adv. 2022;54:107822.
37. Yu S, Zheng C, Zhou F, Baillie DL, Rose AM, Deng Z, et al. Genomic identification and functional analysis of essential genes in Caenorhabditis elegans. BMC Genomics. 2018;19(1):1–14.
38. dos Santos G, Schroeder AJ, Goodman JL, Strelets VB, Crosby MA, Thurmond J, et al. FlyBase: introduction of the Drosophila melanogaster Release 6 reference genome assembly and large-scale migration of genome annotations. Nucleic Acids Res. 2015;43(D1):D690–D7.
39. Birney E, Andrews TD, Bevan P, Caccamo M, Chen Y, Clarke L, et al. An overview of Ensembl. Genome Res. 2004;14(5):925–8.
40. Campos TL, Korhonen PK, Hofmann A, Gasser RB, Young ND. Combined use of feature engineering and machine-learning to predict essential genes in Drosophila melanogaster. NAR Genom Bioinform. 2020;2(3):lqaa051.
41. Howe KL, Bolt BJ, Cain S, Chan J, Chen WJ, Davis P, et al. WormBase 2016: expanding to enable helminth genomic research. Nucleic Acids Res. 2016;44(D1):D774–D80.
42. Campos TL, Korhonen PK, Sternberg PW, Gasser RB, Young ND. Predicting gene essentiality in Caenorhabditis elegans by feature engineering and machine-learning. Comput Struct Biotechnol J. 2020;18:1093–102.
43. Zhang R, Ou HY, Zhang CT. DEG: a database of essential genes. Nucleic Acids Res. 2004;32(suppl1):D271–D2.
44. Rahman MS, Aktar U, Jani MR, Shatabda S, iPromoter-FSEn. Identification of bacterial sigma(70) promoter sequences using feature subspace based ensemble classifier. Genomics. 2019;111(5):1160–6.
45. Shrikumar A, Prakash E, Kundaje A. GkmExplain: fast and accurate interpretation of nonlinear gapped k-mer SVMs. Bioinformatics. 2019;35(14):i173–i82.
46. Campos TL, Korhonen PK, Young ND. Cross-Predicting Essential Genes between Two Model Eukaryotic Species Using Machine Learning. Int J Mol Sci. 2021;22(10).
47. Pradhan UK, Meher PK, Naha S, Pal S, Gupta A, Parsad R. P l DBPred: a novel computational model for discovery of DNA binding proteins in plants. Brief Bioinform. 2023;24(1):bbac483.
48. Xiao Q, Wang J, Peng X, Wu F-x, Pan Y, editors. Identifying essential proteins from active PPI networks constructed with dynamic gene expression. BMC genomics. Springer; 2015.

Figures

Figure 1

Graph structure of gene sequences

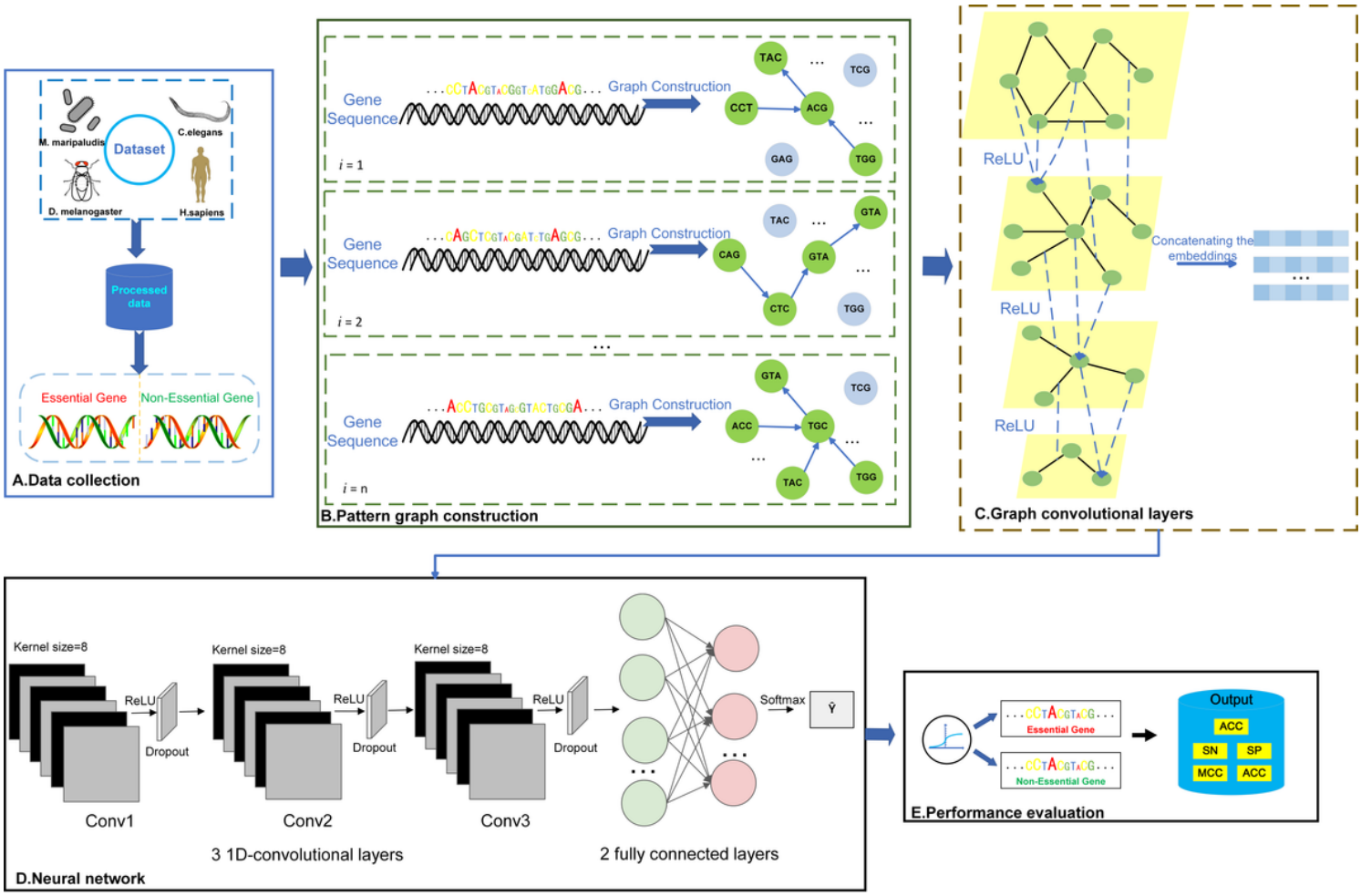

Figure 2

Model GCNN-SFM predicts the structural flow of essential genes

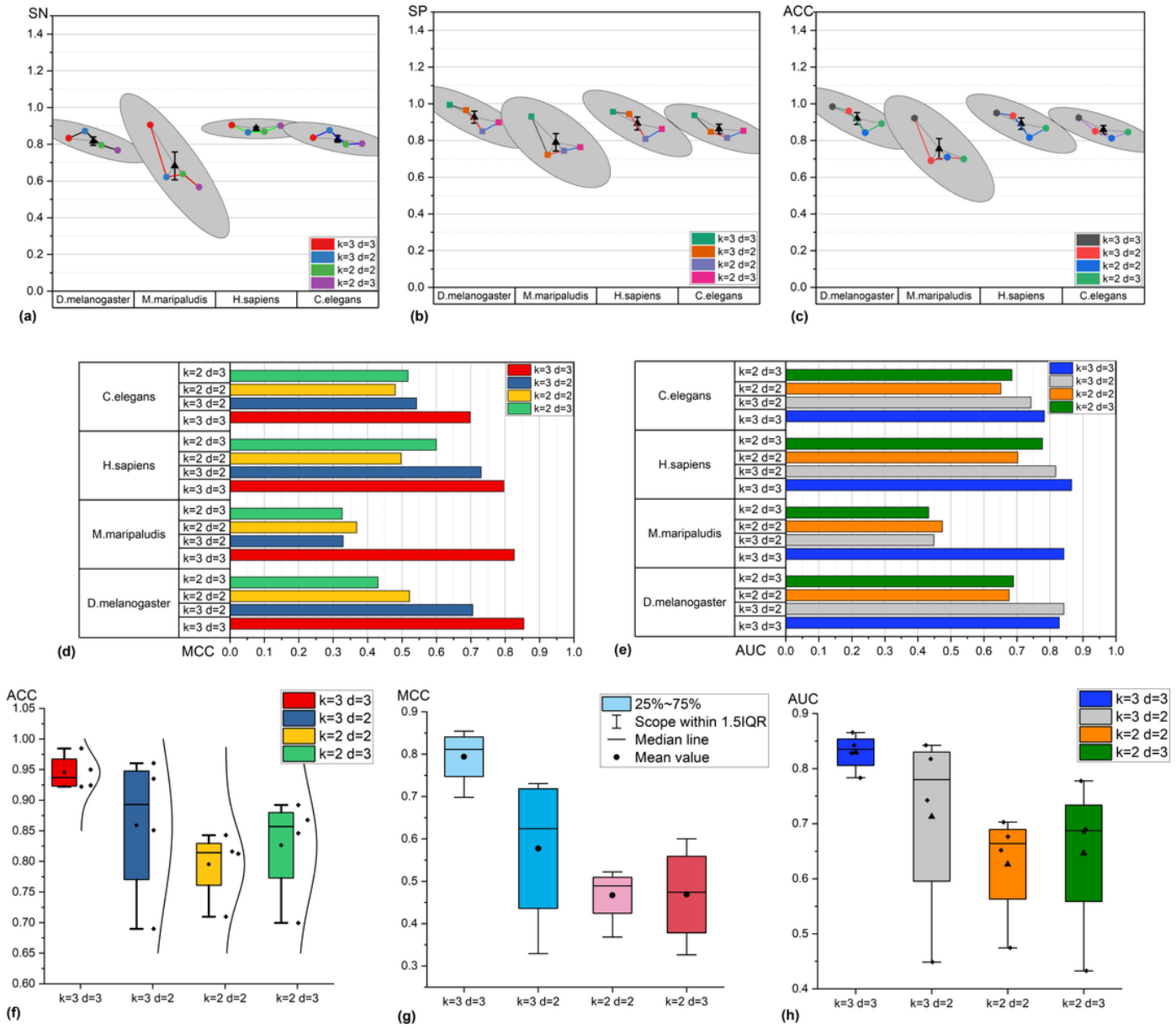

Figure 3

Comparison of performance results of independent datasets testing graph coding methods with different parameters

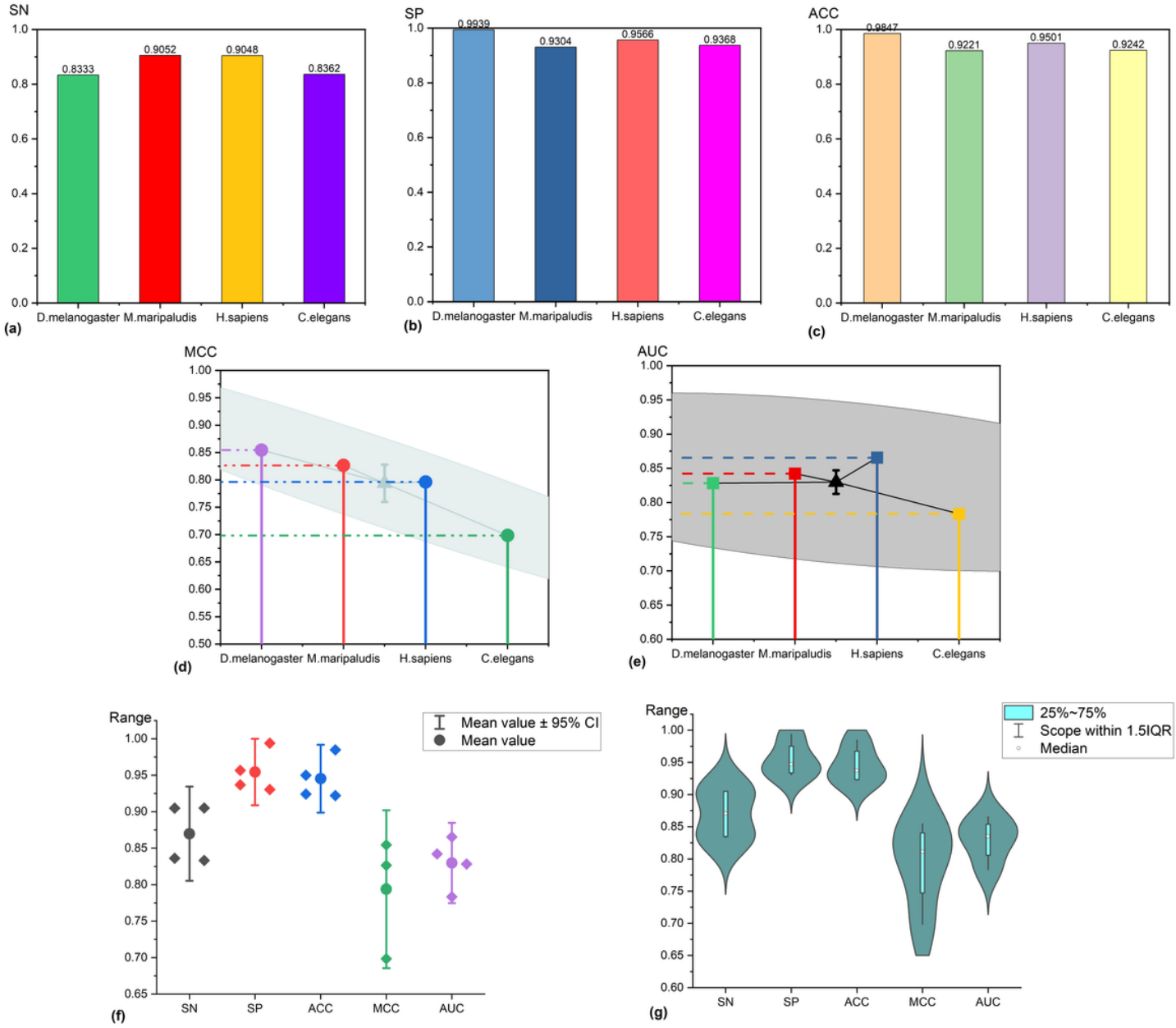

Figure 4

Performance results of different independent datasets testing the essential gene prediction model.

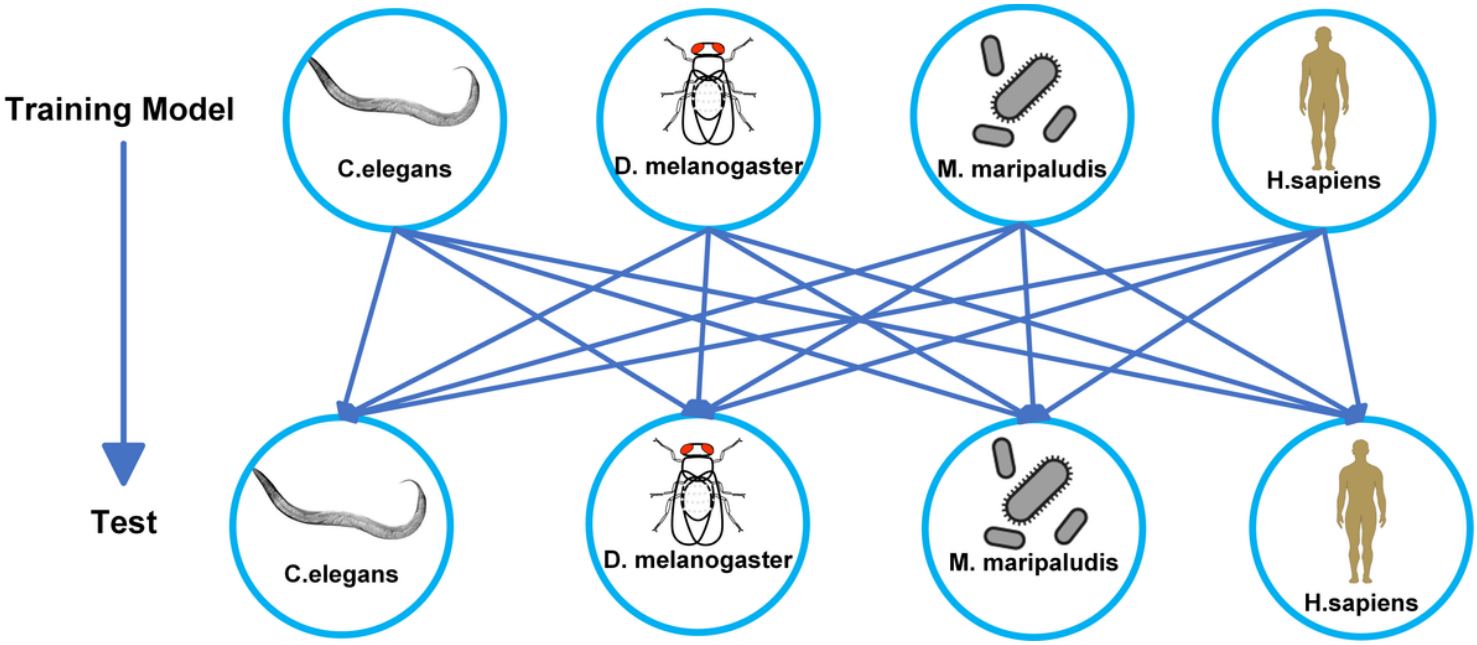

Figure 5

Cross-training of datasets from different species

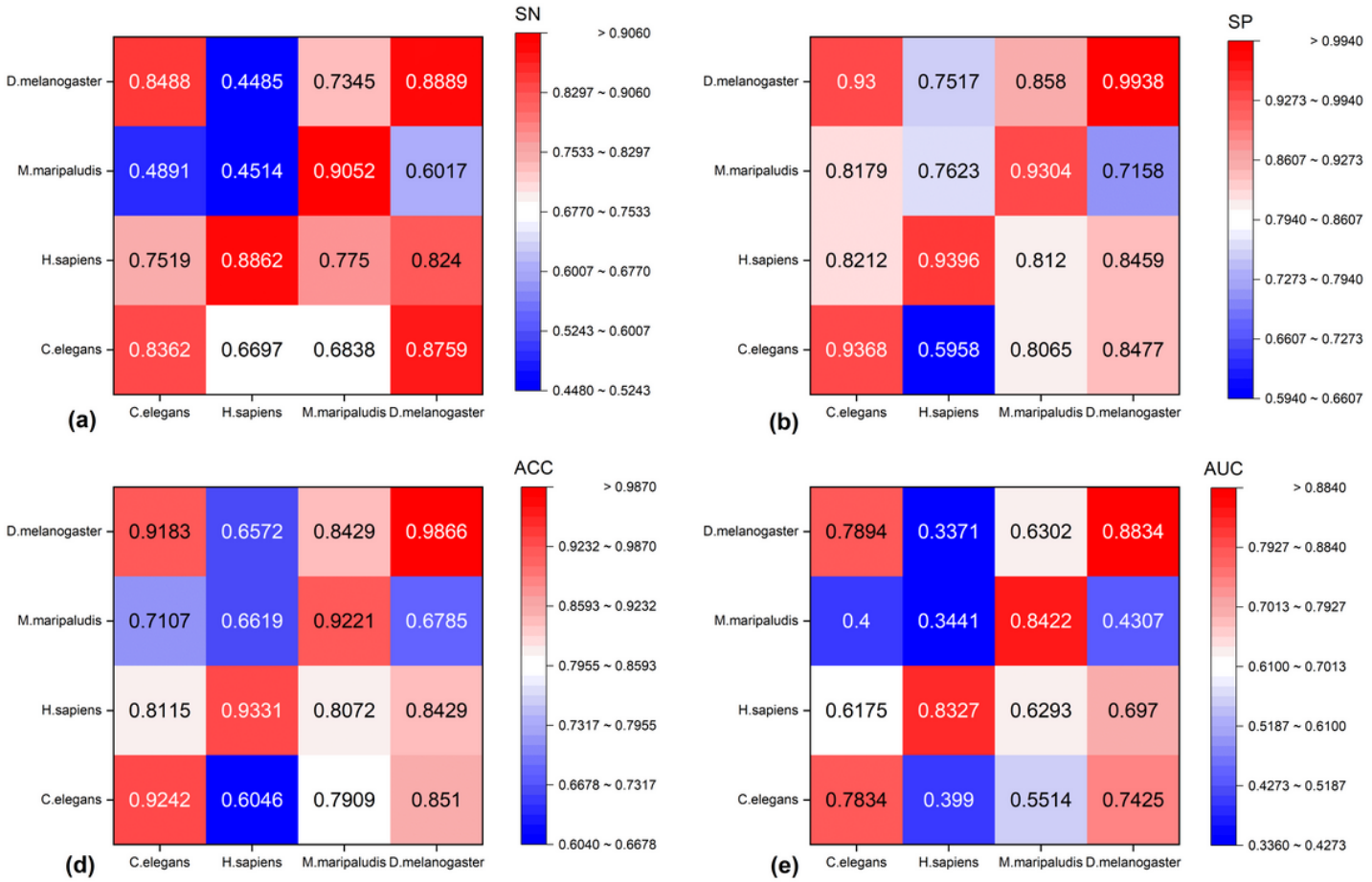

Figure 6

Performance comparison of model validation across species

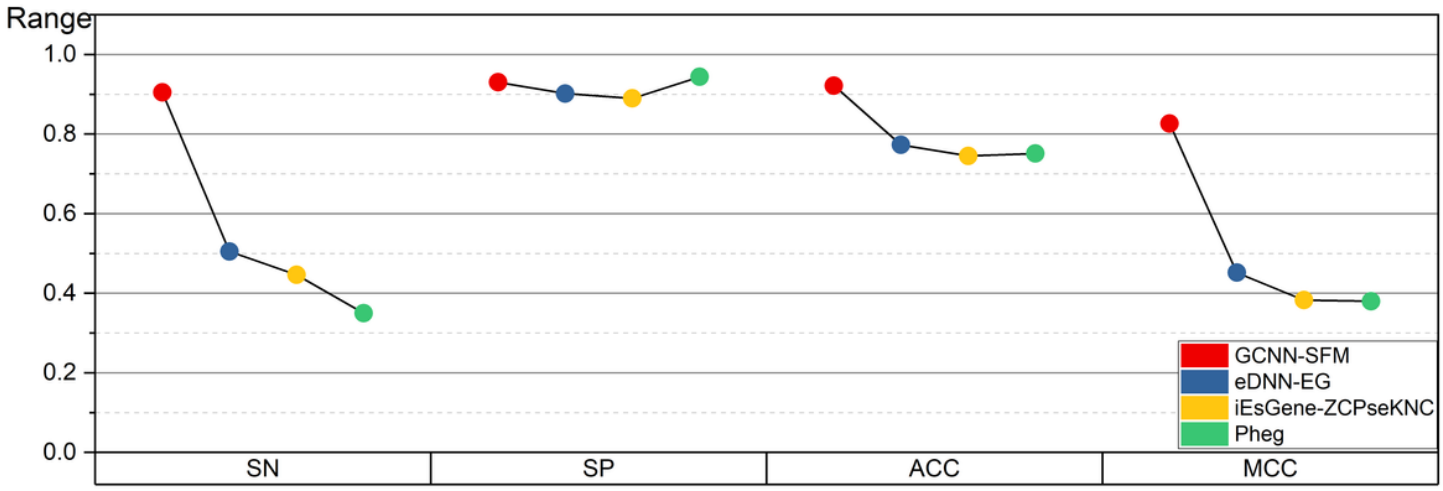

Figure 7

Performance comparison of GCNN-SFM with other existing models

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Graphicabstract.png