International Journal of Advanced Culture Technology Vol.10 No.2 201-212 (2022)
DOI https://doi.org/10.17703/IJACT.2022.10.2.201

IJACT 22-6-27

AllEC: An Implementation of Application for EC Numbers Prediction based on AEC Algorithm

[1]Juyeon Park, [1]Mingyu Park, [2]Sora Han, [3]Jeongdong Kim, [2]Taejin Oh, [3]Hyun Lee*

[1]Dept. of Computer and Electronic Engineering, Sunmoon University, 70 Sunmoonro 221, Tangjeong-myeon, Asan-si, Chungnam 31460, Korea
[2]Dept. of Life Science and Biochemical Engineering, Graduate School, Sunmoon University, 70 Sunmoonro 221, Tangjeong-myeon, Asan-si, Chungnam 31460, Korea
[3]Prof., Div. of Computer Science and Engineering, Sunmoon University, 70 Sunmoonro 221, Tangjeong-myeon, Asan-si, Chungnam 31460, Korea
*juyeone125@gmail.com, duveen@sunmoon.ac.kr, 553sora@naver.com, kjd4u@sunmoon.ac.kr, tjoh3782@sunmoon.ac.kr, mahyun91@sunmoon.ac.kr**

Abstract

With the development of sequencing technology, there is a need for technology to predict the function of the protein sequence. Enzyme Commission (EC) numbers are becoming markers that distinguish the function of the sequence. In particular, many researchers are researching various methods of predicting the EC numbers of protein sequences based on deep learning. However, as studies using various methods exist, a problem arises, in which the exact prediction result of the sequence is unknown. To solve this problem, this paper proposes an All Enzyme Commission (AEC) algorithm. The proposed AEC is an algorithm that executes various prediction methods and integrates the results when predicting sequences. This algorithm uses duplicates to give more weights when duplicate values are obtained from multiple methods. The largest value, among the final prediction result values for each method to which the weight is applied, is the final prediction result. Moreover, for the convenience of researchers, the proposed algorithm is provided through the AllEC web services. They can use the algorithms regardless of the operating systems, installation, or operating environment.

Keywords: *Bioinformatics, Amino Acid Sequence, Enzyme Commission Number, Function Prediction, All Enzyme Commission*

1. INTRODUCTION

The continuous evolution of genomic sequencing developments can rapidly identify large lists of genes from various organisms. The biology researcher needs to closely approach the characterized organisms by identifying functional roles for individual genes within that gene list through sequencing results [1]. For this reason, some applications for the recognition and classification approaches to the characterized organisms have increased in recent years. In general, the manually annotated and reviewed work is time-consuming and expensive, whereas automatically annotated and computational methods help cover as much as possible using

Manuscript received: April 18, 2022 / revised: May 20, 2022 / accepted: June 2, 2022
Corresponding Author: mahyun91@sunmoon.ac.kr
Tel: +82-041-530-2218, Fax: +82-041-530-2218
Professor, Div. of Computer Science and Engineering, Sunmoon Univ., Korea

various calculation formulas quickly [2]. Moreover, before starting an experiment, researchers prefer the computational approach because it can reduce the costly and time-consuming direction of experiments. One of the main goals of the computational approach is to use the automatically annotated method to predict enzyme classes of protein functions in about one sequence. To date, there are more than 4,000 protein sequences of unknown function, and the number continues to increase. However, their ability to assign specific tasks to nucleotide sequences or amino acid sequences is much lower than to classify or identify groups for protein families [3].

Among them, the prediction of Enzyme Commission (EC) numbers is one of the ways to know protein function. Identifying an enzyme function is classifying enzymes according to the reactions they catalyze through four digits of EC numbers. Each position uses a dot to represent four digits and is computationally stored within a unique ontology term by the Nomenclature Committee of the International Union of Biochemistry. EC numbers consist of numbers in each part, and this means they correspond to class, subclass, sub-subclass, and serial number, respectively (i.e., 1.1.1.1) [4]. An EC number having all four numbers in each position about a protein sequence means the most specific annotation for enzyme feature prediction, which allows for associating the protein sequence with specific chemical reactions [5]. For example, the enzyme's function of Myeloperoxidase has the code "1.11.1.7"; The first digit (1) denotes oxidoreductase; the second digit (11) acts on peroxide as an acceptor; the third digit (1) shows Peroxidases; and the last number (7) refers a Myeloperoxidase.

Through previous studies, models could predict EC numbers through a computational approach. However, each model provides a downside that the users' expected results differ slightly based on the three main research directions [6]. The three directions of the computational approach that users should consider are summarized as follows. Firstly, some researchers believed that protein structures determine function. They made methods that could predict the function of enzymes from the enzyme structure, such as Cofactor [7-8], i-tasser [9], etc. Second, the common assumption that a high sequence similarity can have similar functions is currently the most used for predicted enzyme function. EFICAz [10-12] was a representative method made using high sequence similarity. Thirdly, the most used approach is using machine learning algorithms to extract features from sequences and predict enzymes. They have recently been developed such as MF-EFP [13], DeepEC [14], and DEEPred [15], by machine learning algorithms.

However, despite many of these models studied, direct users (the biologists or bio researchers) find it challenging to use the computer. Some models provide a website, but most models require a download after creating a specific environment on a computer. Therefore, we developed AllEC web service, which allows users to easily predict EC numbers by linking some computational models and presenting the results within one website. When the user inputs one amino acid sequence into the AllEC web service, the results of each model are displayed at once, and the results are shared with the user, providing as many options as possible for protein prediction results. The results of the AllEC web service can provide more accurate EC number information to users. It can predict the EC number with higher accuracy than the existing method and save time and cost.

2. RELATED WORK

Most EC number prediction methods require downloading after creating a specific environment on the computer. In this process, users (the biologists or bio researchers) experience a lot of confusion because they are not familiar with the programs. In addition, the prediction results for each model are different, which can lead to user confusion. Moreover, each model has a different method of predicting EC numbers, and there is a gap in the number of EC numbers that each model can cover. Therefore, the predicted values of each model are naturally different. AllEC web services can help popular prediction methods for EC number prediction

design be used on the web to reduce confusion. The four methods (tools) we used have the following characteristics.

2.1 DeepEC

DeepEC [14] is a deep learning-based method that automatically predicts 4-digit EC numbers. It reported a deep-learning-based computational framework that predicts EC numbers for protein sequences with high precision and in a high-throughput manner. This approach uses three convolutional neural networks (CNNs) as a significant engine for predicting EC numbers. In addition, it implements homology analysis for EC numbers that the CNNs cannot classify, and can cover a total of 4,669 EC numbers prepared by processing protein sequences.

2.2 DETECTv2

DETECTv2 [16-17] is an enzyme annotation method that considers the impact of sequence diversity when assigning enzyme functions to protein sequences by EC number. This approach constructs positive and negative density profiles for each EC number, representing how sequences within a class align with sub and sub-sub classes. It can provide more enzyme classes than previous versions, significantly increasing the precision and recall of assignments and providing specific cut-offs that show performance in the context of pathways. It can cover a total of 786 EC numbers prepared by processing protein sequences.

2.3 ECPred

ECPred [18] is a probabilistic EC number prediction method that researchers can use on the web. It proposes a novel enzymatic function prediction method based on the ensemble of machine learning classifiers. First, the model predicts whether the query sequence is enzymatic or non-enzymatic and has structures representing the major EC classes. After deciding the main EC class of a query, subclass, sub-subclass, and substrate classes predicted. Therefore, it provides predictions for 858 EC numbers, including 6 main classes, 55 subclass classes, 163 sub-subclass classes, and 634 substrate classes.

2.4 eCAMI

eCAMI [19] was also generalized to extract characteristic k-mer peptides for all the Swiss-Prot database enzymes classified by the EC numbers and applied to enzyme EC prediction. This approach implements as a Python package for EC classification and annotation. In addition, it can provide the best accuracy and memory for Carbohydrate active enzyme (CAZyme) and enzyme EC classification and annotation. The approach is specialized to the CAZyme family, so predicting the general EC number was difficult. Therefore, it is excluded from the model comparison. However, it is recommended to use this model to indicate the EC number related to CAZyme.

3. METHOD

We developed the AllEC web service to improve the existing complex EC number prediction process. The AllEC web service shows the prediction results using the previously widely used EC number prediction methods. In addition, we propose an All Enzyme Commission (AEC) algorithm that recommends the most accurate EC number through the results of each prediction method.

3.1 AEC Algorithms

Owing to the importance of the prediction function to a protein, newly sequenced genomes or an amino acid sequence are often predicted by EC numbers. While many have a single predictive method, we propose AEC, an algorithm that synthesizes the results of existing EC number predictive methods and recommends the most optimal EC numbers. These studies choose an automated EC number-based enzymatic function prediction method that takes the amino acid sequences as inputs. Furthermore, the four methods selected can be a method that can accurately predict four digits EC numbers according to their respective method feature and consider the role of multiple function enzymes.

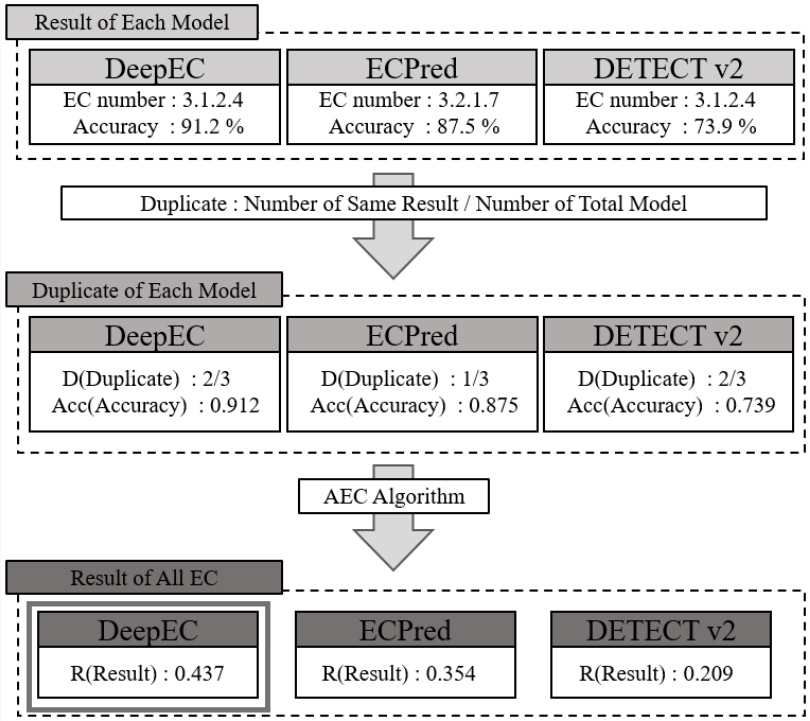

Figure 1. Prediction Process of according to AEC algorithms

The AEC is calculated in the following way. First, we check the duplication (D) of the three EC number prediction results. Here, D means the numerical value of whether there are overlapping EC numbers among the results of each method. After checking the duplication of each method, we compare the accuracy. The formula for multiplying the accuracy by D to give weight and making the sum of each result equal to 1 is as follows. Here, D means the sequence number for classifying each method, n means the total number of methods, and Acc means the accuracy of each method. We checked whether the programs are executable in advance so that each program can be executed. The final prediction is calculated based on the above AECi formula, and the process for the AEC algorithm is shown in Figure 1.

$$\mathrm{AEC_i} = \frac{\mathrm{D_i} \times \mathrm{Acc_i}}{\sum_{k=1}^{n}(\mathrm{D_k} \times \mathrm{Acc_k})} \qquad (1)$$

Therefore, predicting EC numbers through AEC algorithms using accuracy and duplication provides more optimal prediction results.

Algorithm 1: Calculation of the *AEC*

D: duplicate, ***N***: number of methods, ***R***: accuracy with duplicate applied
Input: ***EC***: list of *ec* prediction result of each method, ***ACC***: list of the accuracy of each method

function *CalculateAEC* (*EC*, *ACC*)
  for each *ec* **in** *EC*
    D_i = **count** *ec* **in** *EC* / *N*
  end for
  for *i* = 1 **in** *N*
    $R_i = D_i \times ACC_i$
  end for
  R = **sum**(R_1, R_2, …, R_N)
  $AEC_i = R_i / R$
  i = **index** (**max** (*AEC*))
  return EC_i
end function

Algorithm 2: Processing for predict *ec* number using *AEC*

EC: list of *ec* prediction result of each method, ***ACC***: list of the accuracy of each method
Input: ***S***: amino acid sequence, ***T***: list of methods

function *PredictEC* (*S*)
  for each *t* **in** *T*
    create *Process* (*t*, *S*)
    run *Process*
  end for
  wait all processes when finished
  for *i* = 1 **in** *N*
    EC_i = *ec* of R_i
    ACC_i = *acc* of R_i
  return *CalulateAEC* (*EC*, *ACC*)
end function

Algorithm 1 is a method to obtain the AEC for each prediction method and calculates the AEC value using the EC number and accuracy value predicted by each method. First, calculate the duplicate value of the EC number predicted by each method. Calculate the R for each method by multiplying the duplicate value and the accuracy value. Multiply the sum of R for each method and R for all methods to calculate the AEC. Find the largest value among the AECs for each method, and return the EC number predicted by the method as the final result.

Additionally, algorithm 2 shows the whole flow for predicting EC numbers. For each method, create a subprocess and start prediction, and wait until all subprocesses are finished. Then, the EC and ACC are obtained from the results for each method and stored in a list. With all predicted ECs and ACCs, the final EC is obtained using the CalculateAEC function.

3.2 AllEC Web Services for EC Numbers Prediction

This work addresses AllEC web service prediction; Figure 2 illustrates the development of the AllEC process for EC number prediction from the amino acid sequence.

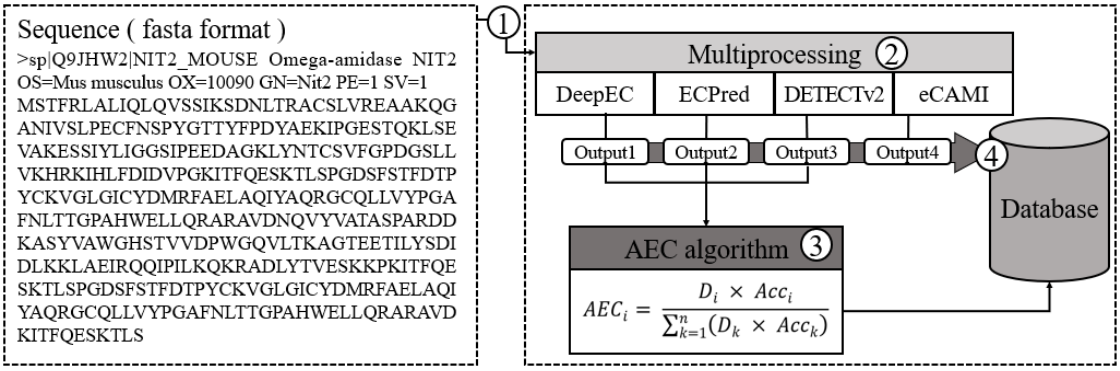

Figure 2. Process of prediction for AEC

Step-1: EC number prediction begins with the user entering the amino acid sequence.

Step-2: Next, the model creates a process to run prediction methods using the multi-processing module we provide to shorten work time.

Step-3: When the prediction of each process is completed, the prediction result is used as a parameter value of the AEC algorithm. When the EC number prediction through the AEC algorithm comes out, five results are provided to the user, including the result of each method and the result of the AEC algorithm.

Step-4: When providing EC number information, refer to the enzyme information in the BRENDA database(https://brenda-enzymes.org/), the enzyme information database system mentioned above, to show the accepted name and reaction information for the corresponding EC number.

4. EXPERIMENTS

4.1 Experiment Environment

The AllEC web services use MariaDB, a type of RDBMS (Relational Database Management System). It is lighter and faster than MySQL, is cost-free, and is easily accessible to researchers using the GPL license. Enzyme information was used by obtaining EC numbers and other information from BRENDA (https://brenda-enzymes.org/, Enzyme Information Database System), one of the most comprehensive enzyme storages.

This paper proposes the AEC algorithm-based AllEC web service, which can execute four existing EC number prediction methods at once and predict the optimal EC number. The AEC algorithm used the duplication and accuracy of the prediction results of DeepEC, ECPred, and DETECTv2, excluding eCAMI, which does not provide accuracy. Here, to confirm that the AEC algorithm used to provide the best EC number improved over the existing system, these results compared the accuracy of AEC and four prediction methods. The evaluation environment runs on i9-10940x, 256GB, RTX 2080Ti.

4.2 Performance of each method

Accuracy refers to the ratio of how many samples are predicted correctly by AEC among the total number of samples. The formula for calculating accuracy is as follows:

Whether the AllEC web service provides the correct is check to EC number, accuracy calculated and compared using 6,091 sample sequences. These sequences were selected as satisfying the following four conditions:

- A sequence whose protein existence is evident at the protein level to select an annotation sequence with a high probability of accuracy;

- In the case of ECPred and DETECTv2, an EC number with a main class 7 is not supported, so sequences excluding main class 7;
- When the length of the sequence is 300-400 characters to select an appropriate length sequence for a smooth test;
- Multiple enzymes that can affect the precision value are excluded.

We compared the four methods and the AEC algorithm through accuracy, precision, recall, and F-1 score. Performance evaluation measures of each model are shown in Table 1. It shows the highest probability among the three models: DeepEC shows an accuracy of 73.42%, and the AEC algorithm shows 75.42%. Although the Precision value is lower than DeepEC, it can be seen that the F-1 score, which harmonizes both precision and reproducibility, is higher than DeepEC.

Table 1. Performance evaluation measures of AEC and three methods

Methods	Accuracy	Precision	Recall	F-1 score
AEC	**75.42**	**50.88**	**51.67**	**50.36**
DeepEC	**73.32**	**51.84**	**50.36**	**50.31**
ECPred	**44.46**	**18.94**	**18.04**	**18.35**
DETECTv2	**46.99**	**19.76**	**19.02**	**19.32**

Table 2 shows an example of recommending the correct EC number using the AEC algorithm when each model's EC number prediction results are different. The comparison of prediction results is a total of 6,091 queries according to four methods. These prediction results can be gathered to increase the overall prediction success rate when predicted using only one method. For example, for sp|P9WQB9|ADHD MYCTU data, DeepEC predicted 1.1.1.90, but AEC predicted 1.1.1.1 because DETECTv2 is more likely to be 1.1.1.1.

Table 2. Comparison of prediction results

Query ID	True	AllEC	DeepEC	ECPred	DETECTv2
sp|P9WQB9|ADHD MYCTU	**1.1.1.1**	**1.1.1.1**	**1.1.1.90**	**N/A**	**1.1.1.1**
sp|Q9JYY8|RLMK_NEIMB	2.1.1.264	2.1.1.264	2.1.1.264	**N/A**	**2.5.1.47**
sp|Q4WMJ8|GLIJ_ASPFU	3.4.13.19	3.4.13.19	3.4.13.19	**N/A**	**N/A**
sp|Q5JJ82|MFNA_THEKO	4.1.1.11	4.1.1.11	4.1.1.11	4.1.1.11	**N/A**
…	…	…	…	…	…
sp|Q9QZS5|SGK2_MOUSE	2.7.11.1	2.7.11.1	**N/A**	2.7.11.1	**4.1.2.13**

5. IMPLEMENTATION

5.1 Implementation Environment for AllEC Web Service

In this research, we built an environment for running each prediction method (tools). For example, DeepEC and eCAMI require Python version 3.6, ECPred requires Java 8.0, and DETECT v2 requires Python 2.7. For these methods, we built the server with Ubuntu 18.04 because each device was executing in a Linux operating system. Moreover, each program requires BioPython [20], a free-to-use tool for biological calculations; NCBI-BLAST [21], a sequence similarity search tool; and EMBOSS, an open-source software package for biology. When trying to run a web service, it is necessary to check whether the programs are executable in advance so that each program can be running. If not installed, it automatically sets the execution environment. Further, we minimized mutual interference by setting up a virtual environment so that the execution conditions of each

method did not conflict with one another.

5.2 Results of Implementation for AllEC Web Service

This paper provides a web service that allows biologists to predict EC numbers easily. To do this, we built a user interface through the Tabler UI framework (https://github.com/tabler/tabler) based on bootstrap. It can be used by anyone and is freely available (MIT license). When predicting the EC number in the AllEC web service, the user can directly enter the amino acid sequence in FASTA format or attach a file. Figure 3 shows an example of input format on the AllEC web services.

At this point, a sample sequence is provided for users to understand the AllEC web service easily. The user can choose whether to receive the prediction result by e-mail and press the submit button to start the prediction. After the forecast is complete, the work is shown on Predict History page. After the prediction is achieved, the result is output. When the Show Logs button is pressed, the prediction log of each tool is provided. When predicting the EC number, the entered amino acid sequence and the result are automatically stored in the database. If the user is logged in, the users can view their prediction history. In addition, the AllEC web services (https://allec.cpslab.tech/) provides an EC number search against the Brenda database on the EC Tree page. Here, you can check the EC number, accepted name, and reaction information. The user can search the stored contents, and the information to be output at this time is the Job ID given at the time of prediction, Query ID of each sequence, and date information. At this time, if the user presses the Detail button, they can see the screen that was output at the time of prediction again. The user can search the stored contents, and the information to be output at this time is the Job ID given at the time of prediction, Query ID of each sequence, and date information. Then, if the user presses the Detail button, they can see the screen that was output at the time of prediction again (Figure 3-4).

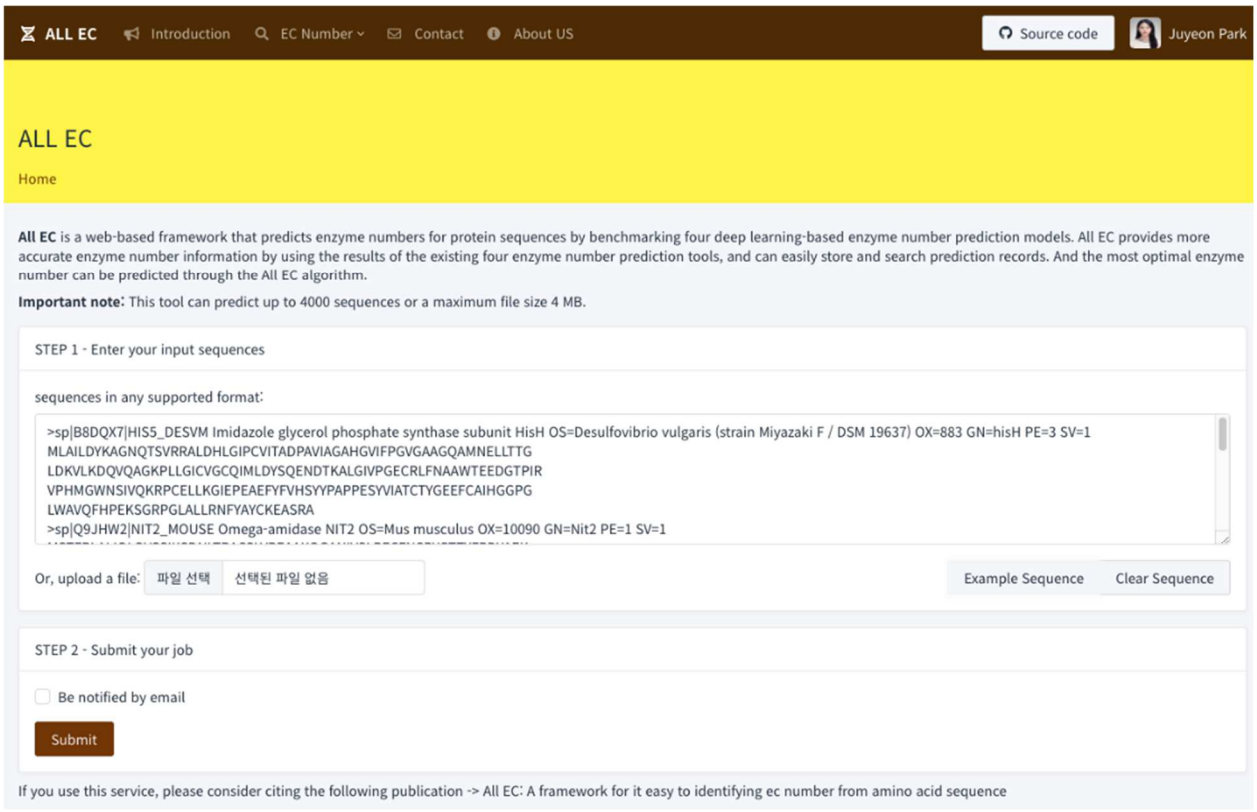

Figure 3. Snapshot of the AllEC web service for input amino acid sequence

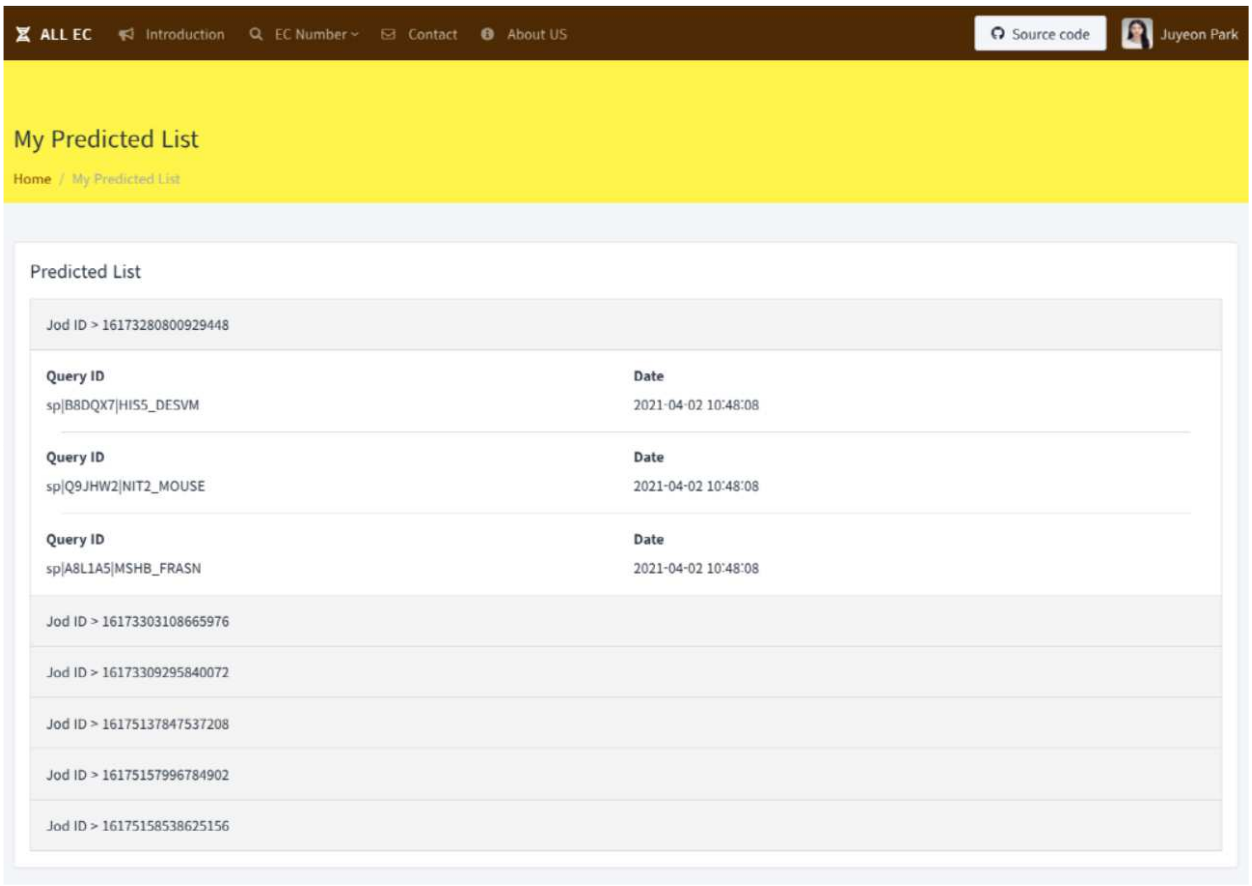

Figure 4. Snapshot of the AllEC web service for prediction lists

Figure 5 shows a snapshot of the prediction list from AllEC web service. The prediction list includes all information on EC numbers.

(A) The result screen first outputs the Job ID automatically assigned when predicting the EC number, the predicted time, and the input sequence.
(B) The result of the Recommend algorithm is output at the top, and the output information includes the Query ID, Description, and EC number of the input sequence, Accuracy, Accepted name, and Reaction information.
(C) The prediction results of DeepEC, ECPred, DETECTv2, and eCAMI are all output.
(D) To check the predicted log, click the Show Log button.

The AllEC web service provides a service for predicting EC numbers for amino acid sequences. The AllEC web service uses Flask (Part of micro web framework) based on the Werkzeug toolkit and jinja2 template engine. It does not require any special methods or libraries. Flask supports extensions to add application functions as implemented by itself, so it is optimal for handling object relationship mappers, form validation, upload management, various open authentication technologies, and methods related to several common frameworks. To help users with no programming experience, we have developed a web service that allows users to submit amino acid sequences and get EC numbers recommended by AEC according to server recommendations. The AllEC web service provides source code about EC number prediction, enzyme information table, and EC number prediction records at GitHub (https://github.com/sm-cps-lab/AllEC.git), and

the web applications can be used for free at https://allec.cpslab.tech/.

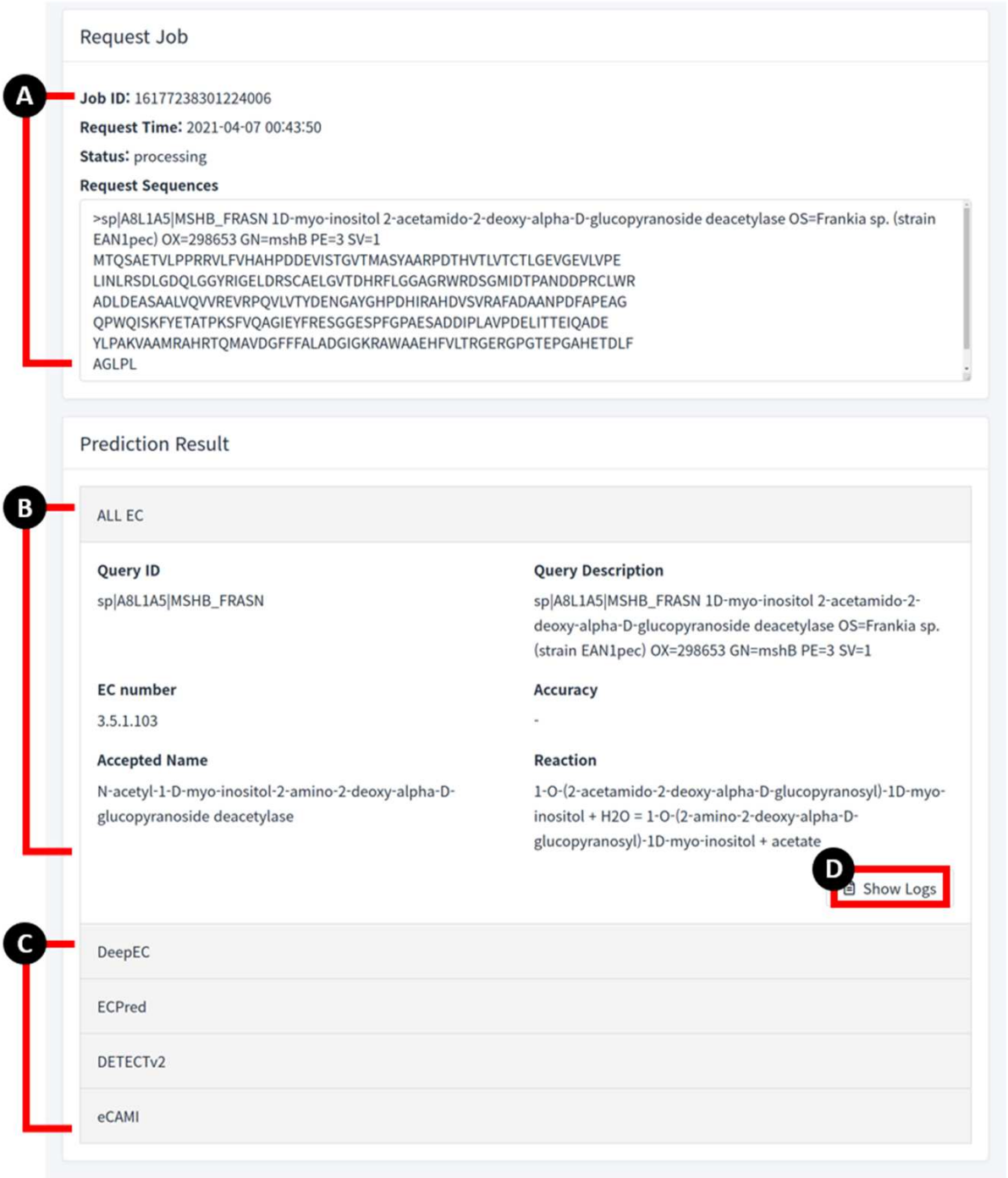

Figure 5. Snapshot of the AllEC web service for Prediction Result

6. CONCLUSION

In this paper, we proposed an AEC algorithm that integrates the results of various methods to solve the problem of not knowing the exact prediction result when predicting a sequence using several different models. It is a method of selecting the final result by giving each result a duplicate weight. As can be seen from the

experimental results, it shows better results than other prediction tools. These results will help many biologists to direct their functional experiments on protein sequences. Moreover, for the convenience of researchers, algorithms are provided through the AllEC web service. This allows the AEC algorithm to be used regardless of the operating system, installation, or operating environment. Currently, the AEC algorithm derives results using three prediction tools. Future work will predict better results by incorporating more predictive tools.

ACKNOWLEDGEMENT

This research was supported by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education(MOE, Korea) and National Research Foundation of Korea(NRF)

REFERENCES

[1] Kanehisa, M., 2000 KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Research 28: 27–30., doi: 10.1093/nar/28.1.27

[2] Söhngen, C., A. Chang, and D. Schomburg, 2011 Development of a classification scheme for disease-related enzyme information. BMC Bioinformatics 12: 329., doi: 10.1186/1471-2105-12-329

[3] Rose, P. W., A. Prli´c, A. Altunkaya, C. Bi, A. R. Bradley, et al., 2016 42 The RCSB protein data bank: integrative view of protein, gene 43 and 3D structural information. Nucleic Acids Research 45: D271– 44 D281., doi: 10.1093/nar/gkw1000

[4] Cornish-Bowden, A., 2014 Current iubmb recommendations on enzyme nomenclature and kinetics. Perspectives in Science 1: 74–87, Reporting Enzymology Data – STRENDA Recommendations and Beyond., doi: 10.1016/j.pisc.2014.02.006

[5] Webb, E. C., I. U. of Biochemistry, and M. Biology, 1992 Enzyme 63 nomenclature 1992. Recommendations of the Nomenclature Committee 64 of the International Union of Biochemistry and Molecular Biology on 65 the Nomenclature and Classification of Enzymes, Number Ed. 6, 66 Academic Press, San Diego, California, USA.

[6] Li, Y., S. Wang, R. Umarov, B. Xie, M. Fan et al., 2017 DEEPre: sequence-based enzyme EC number prediction by deep learning (J. Hancock, Ed.). Bioinformatics 34: 760–769., doi: 10.1093/bioinformatics/btx680

[7] Roy, A., J. Yang, and Y. Zhang, 2012 COFACTOR: an accurate comparative algorithm for structure-based protein function annotation. Nucleic Acids Research 40: W471–W477., doi: 10.1093/nar/gks372

[8] Zhang, C., P. L. Freddolino, and Y. Zhang, 2017 COFACTOR: improved protein function prediction by combining structure, sequence and protein-protein interaction information. Nucleic Acids Res 45: W291-W299., doi: 10.1093/nar/gkx366

[9] Yang, J., R. Yan, A. Roy, D. Xu, J. Poisson et al., 2014 The I-TASSER Suite: protein structure and function prediction. Nature Methods 12: 7–8., doi: 10.1038/nmeth.3213

[10] Kumar, N., and J. Skolnick, 2012 EFICAz2.5: application of a high-precision enzyme function predictor to 396 proteomes. Bioinformatics 28: 2687–2688., doi: 10.1093/bioinformatics/bts510

[11] Arakaki, A. K., Y. Huang, and J. Skolnick, 2009 EFICAz2: enzyme function inference by a combined approach enhanced by machine learning. BMC Bioinformatics 10: 107., doi: 10.1186/1471-2105-10-107

[12] Tian, W., A. K. Arakaki, and J. Skolnick, 2004 EFICAz: a comprehensive approach for accurate genome-scale enzyme function inference. Nucleic Acids Research 32: 6226–6239., doi: 10.1093/nar/gkh956

[13] Xiao, X., L.-W. Duan, G.-F. Xue, G. Chen, P. Wang et al., 2020 MF-EFP: Predicting Multi-Functional Enzymes Function Using Improved Hybrid Multi-Label Classifier. IEEE Access 8: 50276–50284., doi:

10.1109/ACCESS.2020.2979888.

[14] Ryu, J. Y., H. U. Kim, and S. Y. Lee, 2019 Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. Proceedings of the National Academy of Sciences 116: 13996–14001., doi: 10.1073/pnas.1821905116

[15] Sureyya Rifaioglu, A., T. Doğan, M. Jesus Martin, R. Cetin-Atalay, and V. Atalay, 2019 DEEPred: Automated Protein Function Prediction with Multi-task Feed-forward Deep Neural Networks. Scientific Reports 9:, doi: 10.1038/s41598-019-43708-3

[16] Hung, S. S., J. Wasmuth, C. Sanford, and J. Parkinson, 2010 DETECT—a Density Estimation Tool for Enzyme ClassificaTion and its application to Plasmodium falciparum. Bioinformatics 26: 1690–1698., doi: 10.1093/bioinformatics/btq266

[17] Nursimulu, N., L. L. Xu, J. D. Wasmuth, I. Krukov, and J. Parkinson, 2018 Improved enzyme annotation with EC-specific cutoffs using DETECT v2 (J. Hancock, Ed.). Bioinformatics 34: 3393–3395., doi: 10.1093/bioinformatics/bty368

[18] Dalkiran, A., A. S. Rifaioglu, M. J. Martin, R. Cetin-Atalay, V. Atalay 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., doi: 10.1186/s12859-018-2368-y

[19] Xu, J., H. Zhang, J. Zheng, P. Dovoedo, and Y. Yin, 2019 eCAMI: simultaneous classification and motif identification for enzyme annotation (J. Xu, Ed.). Bioinformatics 36: 2068–2075., doi: 10.1093/bioinformatics/btz908

[20] Cock, P. J. A., T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox et al., 2009 Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics 25: 1422–1423., doi: 10.1093/bioinformatics/btp163

[21] Johnson, M., I. Zaretskaya, Y. Raytselis, Y. Merezhuk, S. McGinnis et al., 2008 NCBI BLAST: a better web interface. Nucleic Acids Research 36: W5–W9., doi: 10.1093/nar/gkn201