# MAKALAH ILMIAH
# PROSIDING SEMINAR INTERNASIONAL
# TERINDEKS SCOPUS

## the International Conference on Climate Change and Sustainability Engineering in ASEAN (CCSE-ASEAN) 2019

Judul:

Sequence-based protein-protein interaction prediction using greedy layer-wise training of deep neural networks

disusun oleh:
Faruq Sandi Hanggara, and Khairul Anam

**JURUSAN TEKNIK ELEKTRO**
**FAKULTAS TEKNIK**
**UNIVERSITAS JEMBER**
**2021**

---

Diseminarkan Universitas Jember, Jember, Indonesia
13 - 14 November 2019

scitation.org/journal/apc

Volume 2278

# Climate Change and Sustainability Engineering in ASEAN 2019

**Jember, Indonesia** • 13 November 2019

**Editors** • Khairul Anam, Retno Utami Agung Wiyono, Rizki Fitria Darmayanti, Felix Arie Setiawan and Abdur Rohman


AIP Conference Proceedings

AIP Publishing

# Preface: Climate Change and Sustainability Engineering in ASEAN 2019

Cite as: AIP Conference Proceedings **2278**, 010001 (2020); https://doi.org/10.1063/12.0000647
Published Online: 26 October 2020

# Preface: Climate Change and Sustainability Engineering in ASEAN 2019

We are delighted to present the Proceedings of the International Conference on Climate Change and Sustainability in ASEAN 2019 (CCSE-ASEAN 2019).

CCSE-ASEAN 2019 was held at the University of Jember, Indonesia, on 13 November 2019. This conference was hosted jointly by the Faculty of Engineering, University of Jember (Indonesia) and the School of Engineering, University of San Carlos (Philippines). The conference will be held annually on alternate cities of Jember (Indonesia) and Cebu (Philippines).

CCSE-ASEAN is an attempt to formulate the best response to climate change that poses increasingly significant threats to sustainable development in Southeast Asia. This conference aims at encouraging rich discussions and continuous collaborations among researchers, engineers, leaders in regional government and industries, and students on enhancing the role of the engineering field with its major innovations in ASEAN countries to mitigate climate change impacts.

CCSE-ASEAN 2019 received 171 submissions of abstracts and full papers. On the basis of a single-blind review process, in which two or three independent reviewers were assigned for each submission, 100 full-papers were accepted for oral presentation. The presenters at CCSE-ASEAN 2019 came from several countries including Indonesia, Philippines, Japan, China, and Iraq. The authors presented original scientific reports on varied topics but highly relevant to climate change and sustainability studies, including new models in disaster management, advances in biomaterials, novel analyses in renewable energy technologies, and uses of artificial intelligence and Internet of Things in farming. Based on further assessment on the overall quality of the presented papers, the CCSE-ASEAN Committee has selected 50 outstanding papers for submission to AIP Conference Proceedings.

Our sincere appreciation goes to all authors who have submitted their abstracts and papers to CCSE-ASEAN 2019 especially to the authors who presented their papers in the parallel session. Our deep gratitude goes to the reviewers for their dedicated work. We sincerely thank Prof. Evelyn Taboada (University of San Carlos, Philippines), Prof. Siti Rozaimah SA (Universiti Kebangsaan Malaysia, Malaysia), Prof. How- wei Chen (National Central University, Taiwan), Dr. Timotius Pasang (Auckland University of Technology, New Zealand), and Dr. Hermann van Radecke (Flensburg University of Applied Science, Germany) for having presented their insightful lectures in the plenary session. We would also thank all committee members of CCSE-ASEAN 2019 for their continuous hard work and cooperation, and we thank our sponsors for their support.

We do hope that all the participants of CCSE-ASEAN 2019 would gain meaningful inspiration and fruitful collaboration from the conference. We also wish them a joyful experience from their stay at Jember during CCSE-ASEAN 2019. We are looking forward to seeing you again in CCSE-ASEAN 2020.

Chair of CCSE-ASEAN 2019
Dr. Eng. Triwahju Hardianto

# Photos: Climate Change and Sustainability Engineering in ASEAN 2019

Cite as: AIP Conference Proceedings **2278**, 010002 (2020); https://doi.org/10.1063/12.0001260
Published Online: 26 October 2020


*Climate Change and Sustainability Engineering in ASEAN 2019*
AIP Conf. Proc. 2278, 010002-1–010002-5; https://doi.org/10.1063/12.0001260
Published by AIP Publishing. 978-0-7354-4008-1/$30.00

# Committee

## *Steering Committee*

**Dean of Faculty of Engineering, University of Jember:** Dr. Ir. Entin Hidayah, M.UM
**Dean of School of Engineering, University of San Carlos:** Prof. Evelyn B. Taboada, Ph.D
**Assistant Dean of the School of Engineering, University of San Carlos :** Engr. Ricardo L. Fornis
**Chair of the Chemical Engineering, School of Engineering, University of San Carlos :** Engr. Luis K. Cabatingan
**Vice Dean of Academic Affairs of Faculty of Engineering, University of Jember:** Dr. Eng. Triwahju Hardianto
**Vice Dean of Financial Affairs of Faculty of Engineering, University of Jember:** Syamsul Arifin, S.T., M.T.
**Vice Dean of Students Affairs of Faculty of Engineering, University of Jember:** Sumardi, S.T., M.T.

## *Organizing Committee*

**General Chair:** Dr. Eng. Triwahju Hardianto
**Vice General Chair :** Luzvisminda M. Bellotindos, Ph.D.

**Vice Chair of CCSE-ASEAN 2019:** Boy Arief Fachri, S.T.,M.T., Ph.D
**Vice Chair of SISTEM 2019:** Dr. Salahuddin Junus, S.T., M.T.
**Vice Chair of KONTSI 2019:** Dr. Anik Ratnaningsih, S.T., M.T.

**Chair of Scientific Committee:** Prof. Dr. Ir. Bambang Sujanarko, M.M.

**Vice Chair of Scientific Committee for CCSE-ASEAN 2019 :** Mahros Darsin, S.T., M.Sc, Ph.D
**Vice Chair of Scientific Committee for SISTEM 2019 :** Dr. Nasrul Ilminnafik, S.T., M.T.
**Vice Chair of Scientific Committee for KONTSI 2019 :** Dr. Gusfan Halik, S.T., M.T.

**Scientific Publication Division**
– CCSE-ASEAN 2019 : Khairul Anam, S.T., M.T., Ph.D
– SISTEM 2019 : Welayaturromadhona, S.Si, M.Sc
– KONTSI 2019 : Willy Kriswardhana, S.T., M.T.

**Funding Manager :** Syamsul Arifin, S.T., M.T.
**Tour Manager :** Dr. Yeny Dokhikah, S.T., M.T.

**Public Relation and IT Division**

– CCSE-ASEAN 2019 : Abdur Rohman, Ph.D, Ivan Augusta Farizkha, M.T., Widya Cahyadi, M.T., Andrita C. Eska, M.T., Dedy W Herdiyanto, M.T., Devita A. Larasati, M.T., Fanteri Aji Dharma, S.T., M.S.
–SISTEM 2019 : Danang Yudistiro, S.T., M.T., Tri Vicca Kusuma Dewi, S.T., M.Sc.
–KONTSI 2019 : Yuliana Sukarmawati, S.T., M.T., Nur Faizin, S.Si, M.Si

# Anam, Khairul

Universitas Jember, Jember, Indonesia

https://orcid.org/0000-0003-4169-2237

Edit profile Set alert Potential author matches Export to SciVal

## Metrics overview

46 Documents by author

320 Citations by **291 documents**

8 *h*-index:

## Document & citation trends


## Most contributed Topics 2015–2019

**Electromyography; Artificial Limbs; Hand Gesture Recognition**
10 documents

**Extreme Learning Machine; Universal Approximation; Moore-Penrose Generalized Inverse**
4 documents

**Motor Imagery; Brain Computer Interface; Visual Evoked Potentials**
1 document

View all Topics

46 Documents | Cited by 291 Documents | 0 Preprints (New) | 42 Co-Authors | Topics

**Note:**
Scopus Preview users can only view an author's last 10 documents, while most other features are disabled. Do you have access through your institution? Check your institution's access to view all documents and features.

Export all Add all to list Sort by: Date (newest)

View list in search results format

View 646 references

Set document alert

Article • Open Access

### Improved myoelectric pattern recognition of finger movement using rejection-based extreme learning machine

0 Cited by

Anam, K., Al-Jumaily, A.

***Telkomnika (Telecommunication Computing Electronics and Control)***, 2021, 19(1), pp. 134–145

View abstract Related documents

Article • Open Access

### Multilayer extreme learning machine for hand movement prediction based on electroencephalography

0 Cited by

Anam, K., Avian, C., Nuh, M.

***Bulletin of Electrical Engineering and Informatics***, 2020, 9(6), pp. 2404–2410

View abstract Related documents

Conference Paper

A Novel Herbal Leaf Identification and Authentication Using Deep Learning Neural Network

0 Cited by

Haryono,, Anam, K., Saleh, A.

***CENIM 2020 - Proceeding: International Conference on Computer Engineering, Network, and Intelligent Multimedia 2020***, 2020, pp. 338–342, 9297952

View abstract | Related documents

Conference Paper

Estimation of Finger Joint Movement based on Electromyography Signal using Long Short-Term Memory

0 Cited by

Anam, K., Avian, C., Swasono, D.I., Muttaqin, A.Z., Ismail, H.

***CENIM 2020 - Proceeding: International Conference on Computer Engineering, Network, and Intelligent Multimedia 2020***, 2020, pp. 86–90, 9298023

View abstract | Related documents

Conference Paper

Voice Controlled Wheelchair for Disabled Patients based on CNN and LSTM

0 Cited by

Sutikno,, Anam, K., Saleh, A.

***ICICoS 2020 - Proceeding: 4th International Conference on Informatics and Computational Sciences***, 2020, 9299007

View abstract | Related documents

Conference Paper • Open Access

Sequence-based protein-protein interaction prediction using greedy layer-wise training of deep neural networks

0 Cited by

Hanggara, F.S., Anam, K.

***AIP Conference Proceedings***, 2020, 2278, 020050

View abstract | Related documents

Conference Paper • Open Access

Development of a visual aid tool for blind people based on faster R-CNN

0 Cited by

Iswahyudi,, Anam, K., Sujanarko, B.

***AIP Conference Proceedings***, 2020, 2278, 020049

View abstract | Related documents

Conference Paper • Open Access

Brushless DC motor speed control based on extreme learning machine (ELM) neural network algorithm

0 Cited by

Ahmadi, S., Anam, K., Sujanarko, B.

***AIP Conference Proceedings***, 2020, 2278, 020014

View abstract | Related documents

Conference Paper • Open Access

Design of electrical wheelchair navigation for disabled patient using convolutional neural networks on Raspberry Pi 3

0 Cited by

Sutikno, S., Anam, K., Sujanarko, B.

***AIP Conference Proceedings***, 2020, 2278, 020048

View abstract | Related documents

Conference Paper • Open Access

Herbal leaf authentication using convolutional neural network on Raspberry Pi 3

0 Cited by

Haryono,, Anam, K., Sujanarko, B.

***AIP Conference Proceedings***, 2020, 2278, 020047

View abstract | Related documents

AIP Conference Proceedings

# Sequence-based protein-protein interaction prediction using greedy layer-wise training of deep neural networks

Cite as: AIP Conference Proceedings **2278**, 020050 (2020); https://doi.org/10.1063/5.0014721
Published Online: 26 October 2020

Faruq Sandi Hanggara, and Khairul Anam

# Sequence-based Protein-Protein Interaction Prediction using Greedy Layer-Wise Training of Deep Neural Networks

Faruq Sandi Hanggara[1)] and Khairul Anam[2)]

*University of Jember, Indonesia*

[1)] Corresponding author: faruqsandi@gmail.com
[2)] khairul@unej.ac.id

**Abstract.** Jamu is an herbal medicine commonly used before the advent of modern medicine. Generally, the herbal formula is obtained empirically and passed down from generation to generation. However, the healing process with herbs is also influenced by such as myths and local customs. This influence causes differences in the use of herbal ingredients to cure the same disease. The result is a collection of herbal recipes that overlap each other without any supporting evidence of its validity. Protein-protein interaction (PPI) is a biological process that is influenced by drugs in the healing process. Therefore, PPI due to the consumption of herbs can be used as evidence of the effectiveness of herbal medicine. PPI analysis needs to be done to study how proteins interact with other proteins. PPI analysis with an experimental method (wet lab) cannot be carried out on extensive data and only covers a portion of protein interaction networks. Therefore, a computational approach needs to be done. In previous studies, predictions of PPIs were proven to be carried out using only protein sequence information. The advantage of using this protein sequence information is that this method is more universal. Information that can be obtained from protein sequences includes Discrete Cosine Transform, Multi-scale Local Descriptor, Autocovariance, and Conjoint Triad. The study with the sequence information has been done using different machine learning approaches, such as Support Vector Machines, Random Forest, and Probabilistic Neural Networks. A deep learning approach has also been done with Stacked-Autoencoder, which tried to construct a hidden structure of protein sequences. Previously, deep learning has also been proven to be able to handle raw and complex data on a large scale and learn the useful and abstract features of perceptual problems such as image recognition and voice. The method proposed in this study is deep neural networks that were trained using stacked-autoencoder and stacked-randomized autoencoder. The extracted features used are conjoint-triad. This study compares both methods which have different characteristics in the construction of layers in deep neural networks. We conducted experiments with k-Fold cross-validation which became the gold standard for most predictive model testing. Our experiments with 5 cross-validations and 3 hidden layers gave an average validation accuracy of 0.89 ± 0.02 for the SAE method and 0.51 ± 0.003 for the ML-ELM.

## INTRODUCTION

Protein-protein interactions are biological processes that are influenced by drugs; therefore, PPI can be an indicator of whether a drug reacts as expected to a disease, or with other drugs. Before the existence of computational methods, research on PPI was carried out by experimental methods in a wet laboratory. A common experimental method is the two-hybrid-based ex vivo and in vitro systems in yeast [1], mass spectrometry [2], dan proteome chip [3]. However, this experimental method approach does not cover a complete PPI network [4] and also labor-intensive work. The approach by combining two-hybrid-based experiments and computational approaches were carried out by [5] and [6]. Therefore, several previous studies have proposed amino acid sequences in proteins as features, and the use of information on protein amino acid sequences has proven to be possible. The features that can be obtained from protein sequences are Discrete Cosine Transform with Substitution Matrix Representation [7], Multi-scale Local Descriptor [8], Autocovariance [9] dan Conjoint Triad [10]. All four are processed using a different computational approach, such as Support Vector Machine [7] [10], Random Forest [8] dan Probabilistic Neural Network [9]. The advantage of using protein sequence information is that this method is more universal [10].

*Climate Change and Sustainability Engineering in ASEAN 2019*
AIP Conf. Proc. 2278, 020050-1–020050-6; https://doi.org/10.1063/5.0014721
Published by AIP Publishing. 978-0-7354-4008-1/$30.00

Deep neural network (DNN) gives satisfactory results on solving problems that require higher abstraction such as image [11] dan speech _recognition [12]. Compared to a single layer feed-forward network (SLFN), DNN can handle raw, complex data and automatically learn useful and more abstract features [13]. But making the network deeper by simply adding hidden layers does not necessarily give the best results.

Deep network training is a difficult task and problems that had been proved effective when applied to shallow networks (SLFN) become inefficient when adapted to deep networks. [14]. Research by [15] also concluded that in general a deep neural network is no better or even worse than a neural network with 1 or 2 hidden layers, and argues that the solution found by gradient-descent learning to the optimum global solution will worsen with increasing network size. A reasonable explanation for this is that optimization based on gradients starting from random initialization allows it to be stuck on local optima. Therefore, the addition of each hidden layer needs to be done constructively [16]. This idea motivated the emergence of ELM stacked-autoencoder and multilayer ideas (which are actually stacked-ELM-autoencoder). Deep networks that use unsupervised pretraining that are done per layer (greedy layer-wise) will position the network in areas in the parameter space where fine-tuning (also called transfer-learning) is avoided from local minima [14].

## METHODS

### Datasets

The list of proteins we use is obtained from Indonesian Herbal Medicine-Herbs Analytics [17], to limit protein in the herbal domain. Then the sequence information for each protein is obtained from UniProt [18], whereas interaction data were obtained from STRING-DB [19]. STRING-DB provides information on interacting protein pairs but does not provide information about protein pairs that do not interact with each other. So we used a list of protein pairs that did not interact with each other from [20]. We get about 68,000 pairs of proteins that interact and 36,000 pairs of proteins that do not interact. Furthermore, under sampling is done so that the data becomes balanced.

#### *Conjoint Triad*

Conjoint Triad (CT) is the method proposed in [10]. CT classifies 3 amino acid sequences in a sequence as a unit called a triad. Amino acids consisting of 20 different types are first grouped into 7 groups by the B3LYP / 6-31G * method and the molecular chemistry approach (Table 1).

**TABLE 1** Amino acid grouping [10]

| Cluster | Amino acid |
|---|---|
| 1 | Ala, Gly, Val |
| 2 | Ile, Leu, Phe, Pro |
| 3 | Tyr, Met, Thr, Ser |
| 4 | His, Asn, Gln, Trp |
| 5 | Arg, Lys |
| 6 | Asp, Glu |
| 7 | Cys |

An example of a protein $P$ that has an amino acid sequence is as follows:

$$P = 3562142411 \quad (1)$$

replaced with:

$$P = 3562142411 \quad (2)$$

Because there are 7 groups, there will be 343 different triads (7 * 7 * 7). The frequency of occurrence of the triad is then stored in the form:

$$V = \begin{Bmatrix} 111 = f1 & 121 = f8 & \cdots 177 = f337 \\ 211 = f2 & 221 = f9 & \cdots 277 = f338 \\ \cdots & \cdots & \cdots \quad \cdots \\ 711 = f7 & 721 = f14 & \cdots 777 = f343 \end{Bmatrix} \quad (3)$$

In the example of protein $P$, the contents of all elements of vector V are 0, except: f276 (*triad* 364), f89 (*triad* 562), f13 (*triad* 621), f149 (*triad* 214), f71 (*triad* 142), f158 (*triad* 424), f23 (*triad* 241), dan f4 (*triad* 411).

## Stacked-Autoencoder (SAE)

Autoencoder is a simple learning circuit that aims to map inputs into outputs with the least amount of distortion possible [21]. Autoencoder is an artificial neural network that applies an unsupervised learning algorithm that concludes a function to construct a hidden structure of data that has no label. [22]. Autoencoder is a neural network with two layers. The first layer is the encoding layer and the second layer is the decoding layer. The number of neurons in the decoding layer is the same as the network input dimension. The purpose of the autoencoder is to calculate $f_{enc}(x) = h$ from an input example $x$, where the $f_{dec}(h)$ can recover $x$ with high accuracy. The objective is to model a two-stage approach to the identity function (Equation 1) with $f_{enc}$ being the encoding function owned by the encoding layer and $f_{dec}$ is a decoding function owned by the decoding layer [14]:

$$f_{dec}(f_{enc}(x)) = f_{dec}(h) = \hat{x} \approx x \quad \text{ii)}$$

An SAE consists of several autoencoder (AE) layers that are alternately trained and the output from the previous layer becomes the input for the next layer. The results of the pretraining stage per layer can be improved by fine-tuning using backpropagation on all layers at the same time.

An example of an SAE with 2 autoencoder layers for the completion of classification with 3 classes can be seen in Figure 4. First, the first layer autoencoder is trained with input data $x_k$ to obtain the $h_k^1$. The $h_k^1$ becomes the first order feature in SAE that can be reconstructed back to $\hat{x}$. Next to get the second-order features $\widehat{h_k^2}$, the second layer autoencoder is trained to use $\hat{x}$ as input and generate code $\widehat{h_k^1}$ After the two autoencoder layers are weighted, fine-tuning is done by utilizing the backpropagation from a softmax classifier layer. The softmax classifier layer uses $\widehat{h_k^1}$ as input to generate the likelihood for each class in the classification (Figure 1).

SAE's natural process of pretraining per layer makes it able to decompose inputs into more abstract features. This refers to the nature of the autoencoder which studies the features that make up the representation of the input. The first layer autoencoder in SAE produces the first-order features, and the second layer autoencoder produces the second-order features that are generated from the first level features, the third layer autoencoder produces the third-order features that are generated from the second level features, and so on. The deeper layers of SAE tend to study higher features.

## Multilayer ELM (ML-ELM)

The nature of ELM is SLFN and is limited to SLFN. Thus, in the past 10 years, most ELM research was generalized SLFN [23]. Therefore, to realize a deep network, the only way is to do layer-by-layer unsupervised learning. According to the theory, ELM is a universal approximator, so it is ELM-autoencoder (ELM-AE). ELM-AE is an ELM that functions like an autoencoder in general. [24] has shown that ELM-AE can be used in unsupervised learning, and is considered to be able to do feature representation similar to single value decomposition (SVD). Unlike SAE which requires fine-tuning after each layer is weighted with the results of autoencoder, Multilayer ELM does not require fine-tuning.

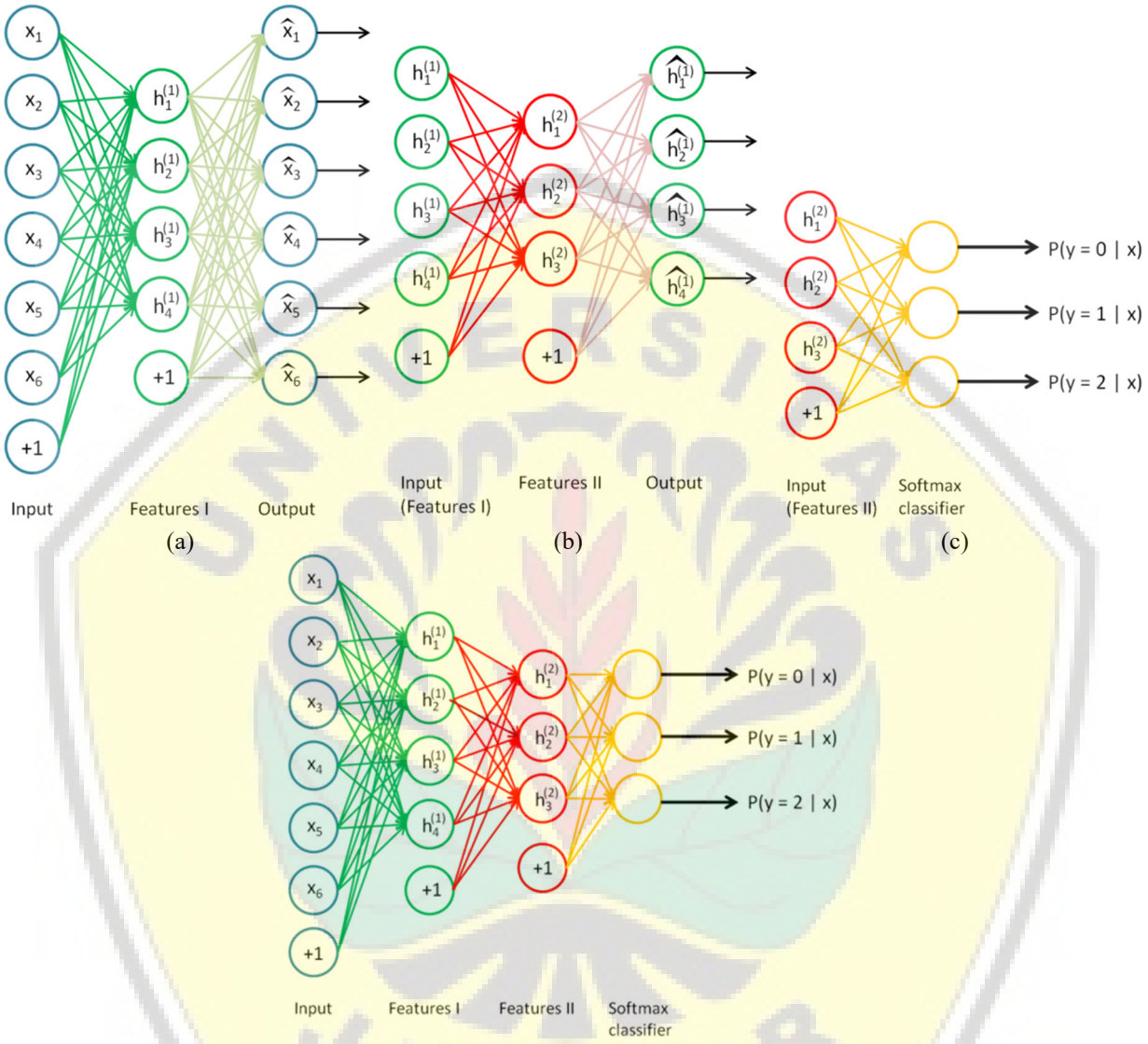


**FIGURE 1**. The architecture of SAE. (a) First AE (b) Second AE (c) Softmax classifier (d) Complete neural networks

## RESULT

We conducted experiments with k-Fold cross-validation which became the gold standard for most predictive model testing. Our experiments with 5 cross-validations and 3 hidden layers gave an average validation accuracy of 0.89 ± 0.020 for the SAE method and 0.51 ± 0.003 for the ML-ELM. Both results are in Table 2, Table 3 and Table 4. We were quite surprised at the performance of ML-ELM, so we continued to inspect the code we use from any mistakes, also using code created by other studies as comparison [24][25]. The results only reached around 0.54 on one occasion or two. However, when we tried to use a single layer ELM, we found the accuracy was slightly higher than 0.7. Then,

a little late, we also found other studies that tested ML-ELM [26]. Single-layer ELM provides more stable accuracy than ML-ELM, and the addition of neurons to the hidden layer in ML-ELM can reduce performance.

**TABLE 2** 5-Cross validation accuracy

| Accuracy | | |
|---|---|---|
| nth-fold | SAE | ML-ELM |
| 1 | 0.90 | 0.51 |
| 2 | 0.91 | 0.52 |
| 3 | 0.84 | 0.52 |
| 4 | 0.90 | 0.51 |
| 5 | 0.89 | 0.51 |
| Avg. | 0.89 | 0.51 |
| Std. | 0.020 | 0.003 |

**TABLE 3** Confusion matrix of SAE result

| | | Actual | | |
|---|---|---|---|---|
| | | Positive | Negative | |
| Pred. | Positive | 27985 | 2775 | 30760 |
| | Negative | 4500 | 29995 | 34495 |
| | | 32485 | 32770 | |

**TABLE 4** Confusion matrix of MLELM result

| | | Actual | | |
|---|---|---|---|---|
| | | Positive | Negative | |
| Pred. | Positive | 15984 | 15307 | 31291 |
| | Negative | 16483 | 17481 | 33964 |
| | | 32485 | 32467 | 32788 |

## CONCLUSION

It is quite ironic to see that deep learning that is supposed to be able to extract features automatically and unsupervised but still requires handmade features. We use CT as a feature because it is easy to create without requiring complex calculations like Autocovariance, so we expect a simple (inexpensive computation time) representation of a sequence as a feature. From this study, we conclude that ML-ELM is not good enough to classify CT features as input. That might be because the feature vector generated by CT does not provide enough variation for ML-ELM. We have tried using the MNIST standard dataset, which gives better results with ML-ELM. For further research, we propose to add fine-tuning to ML-ELM by replacing the last layer with softmax and the like.

## REFERENCES

1. S. Fields and O. K. Song, “A novel genetic system to detect protein-protein interactions,” *Nature*, 1989.
2. A.-C. Gavin *et al.*, “Functional organization of the yeast proteome by systematic analysis of protein complexes,” *Nature*, vol. 415, pp. 141–147, 2002.

3. H. Zhu *et al.*, "Global analysis of protein activities using proteome chips," *Science (80-. ).*, vol. 293, no. 5537, pp. 2101–2105, 2001.
4. J. D. J. Han, D. Dupuy, N. Bertin, M. E. Cusick, and M. Vidal, "Effect of sampling on topology predictions of protein-protein interaction networks," *Nat. Biotechnol.*, vol. 23, no. 7, pp. 839–844, 2005.
5. A. Hin *et al.*, "A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules," *Science (80-. ).*, vol. 295, no. 5553, pp. 321–324, 2009.
6. S. J. Wodak and R. Méndez, "Prediction of protein-protein interactions: The CAPRI experiment, its evaluation and implications," *Curr. Opin. Struct. Biol.*, vol. 14, no. 2, pp. 242–249, 2004.
7. Y.-A. Huang, Z.-H. You, X. Gao, L. Wong, and L. Wang, "Using Weighted Sparse Representation Model Combined with Discrete Cosine Transformation to Predict Protein-Protein Interactions from Protein Sequence," *Biomed Res. Int.*, vol. 2015, pp. 1–10, 2015.
8. Z. H. You, K. C. C. Chan, and P. Hu, "Predicting protein-protein interactions from primary protein sequences using a novel multi-scale local feature representation scheme and the random forest," *PLoS One*, vol. 10, no. 5, pp. 1–19, 2015.
9. Y. Zhao, "Predicting Protein-protein Interactions from Protein Sequences Using Probabilistic Neural Network and Feature Combination," *J. Inf. Comput. Sci.*, vol. 11, no. 7, pp. 2397–2406, 2014.
10. J. Shen *et al.*, "Predicting protein-protein interactions based only on sequences information," *Proc. Natl. Acad. Sci.*, vol. 104, no. 11, pp. 4337–4341, 2007.
11. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.
12. G. Hinton *et al.*, "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *IEEE Signal Process. Mag.*, 2012.
13. Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553. pp. 436–444, 2015.
14. L. Arnold, S. Rebecchi, S. Chevallier, H. Paugam-Moisy, L. Arnold, and S. Chevallier, "An Introduction to Deep Learning," in *European Symposium on Artificial Neural Networks (ESANN)*, 2011.
15. G. Tesauro, "Practical Issues in Temporal Difference Learning," *Mach. Learn.*, vol. 8, no. 3, pp. 257–277, 1992.
16. H. L. Yoshua Bengio, Pascal Lamblin, Dan Popovici, Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy Layer-Wise Training of Deep Networks," *Adv. Neural Inf. Process. Syst.*, 2007.
17. W. A. Kusuma and V. Dewantoro, "Ijah Webserver," 2017. [Online]. Available: http://ijah.apps.cs.ipb.ac.id/.
18. A. Bateman, "UniProt: a worldwide hub of protein knowledge," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D506–D515, Jan. 2019.
19. L. J. Jensen *et al.*, "STRING 8 - A global view on proteins and their functional interactions in 630 organisms," *Nucleic Acids Res.*, vol. 37, no. SUPPL. 1, pp. 412–416, 2009.
20. X. Y. Pan, Y. N. Zhang, and H. Bin Shen, "Large-scale prediction of human protein-protein interactions from amino acid sequence based on latent topic features," *J. Proteome Res.*, vol. 9, no. 10, pp. 4992–5001, 2010.
21. P. Baldi, "Boolean AutoEncoder," pp. 37–50, 2012.
22. T. Sun, B. Zhou, L. Lai, and J. Pei, "Sequence-based prediction of protein protein interaction using a deep-learning algorithm," *BMC Bioinformatics*, vol. 18, no. 1, pp. 1–8, 2017.
23. G. Bin Huang, "What are Extreme Learning Machines? Filling the Gap Between Frank Rosenblatt's Dream and John von Neumann's Puzzle," *Cognit. Comput.*, 2015.
24. L. L. C. Kasun, H. Zhou, G. Bin Huang, and C. M. Vong, "Representational learning with ELMs for big data," *IEEE Intell. Syst.*, 2013.
25. Anton Akusok, Kaj-Mikael Bjork, Yoan Miche, and Amaury Lendasse, "High-Performance Extreme Learning Machines: A Complete Toolbox for Big Data Applic," *IEEE Access*, 2015.
26. D. Xiao, B. Li, and Y. Mao, "A Multiple Hidden Layers Extreme Learning Machine Method and Its Application," *Math. Probl. Eng.*, 2017.