Indonesian Journal of Electrical Engineering and Computer Science
Vol. 18, No. 3, June 2020, pp. 1432~1438
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v18.i3.pp1432-1438

Deep learning in non coding variant (a brief overview)

Lee Kuan Xin, Afnizanfaizal Abdullah
School of Computing (SC), Universiti Teknologi Malaysia (UTM), Malaysia

Article Info

Article history:

Received Jul 28, 2019
Revised Oct 30, 2019
Accepted Nov 13, 2019

Keywords:

Deep learning
Genomics
Neural network
NGS
Non-coding variant

ABSTRACT

The 21st centuries were deemed to be the era of big data. Data driven research had become a necessity. This hold true not only in the business world, yet also in the field of biomedical world. From a few years of biological data extraction and derivation. With the advancement of Next Generation Sequencing, genomics data had grown to become an ambiguous giant which could not keep up with the pace of its advancement in it analysis counter parts. This results in a large amount of unanalysed genomic data. These genomic data consist not only plain information, researcher had discovered the potential of most gene called the non-coding variant and still failing in identifying their function. With the growth in volume of data, there is also a growth of hardware or technologies. With current technologies, we were able to implement a more complex and sophisticated algorithm in analysis these genomics data. The domain of deep learning had become a major interest of researcher as it was proven to have achieve a significant success in deriving insight from various field. This paper aims to review the current trend of non-coding variant analysis using deep learning approach.

Corresponding Author:

Afnizanfaizal Abdullah,
School of Computing (SC), Universiti Teknologi Malaysia (UTM),
81310 Johor Bahru, Johor, Malaysia.
Email: afnizan@utm.my

1. INTRODUCTION

This paper provides the literature review of how deep learning was implemented in non-coding variant studies. It discussed the methods either mathematically or computationally used in identifying and analysing non-coding variant involving next generation sequencing data. In addition, the methods used to analyse non-coding variant will also be included in this chapter.

2. NON-CODING VARIANT

Non-Coding Variants refers to a large category of genomic variants that does not encode protein. Non-coding variants can be either non-coding DNAs or non-coding RNAs(ncRNAs). In an organism, Noncoding DNA sequences are those DNA that does not decode into protein. Meanwhile, non-coding functional RNA should be transcribed from noncoding DNA. Thus, transcripts that are not functional as templates for protein synthesis are called ncRNAs [1]. ncRNAs do not encode proteins but refers as genes that produce functional RNA molecules [2]. Common type of ncRNAs are regulatory ncRNAs and infrastructural ncRNAs [3]. Regulatory ncRNAs can further be classified into distinctive RNA with varying length and structure, namely long non-coding RNAs (lncRNAs), small interfering RNAs (siRNAs, microRNAs (miRNAs) and) Piwi-interacting RNAs (piRNAs). Meanwhile, infrastructural ncRNAs consists of transfer RNAs, ribosomal RNAs, small nucleolar RNAs, and small nuclear RNAs. Additionally, there are enhancer RNAs (eRNAs) and promoter-associated RNAs (PARs) which have been discovered recently. Some common characteristics of Non-coding RNAs RNAs) includes the lack any extensive “Open Reading Frame” (ORF) and a high density of stop codons [4].

2.1. Long ncRNAs

Long ncRNAs are normally longer than 200 nt in length and are mostly non-protein-coding transcripts [5]. The majority of lncRNAs has a low expression level, low level of sequence conservation and nuclear localization due to its high contents of poly A + and poly A2 transcripts. The categorization lncRNAs are ambiguous as it can sometimes be used as transcript template for making short RNAs.

LncRNAs can be classified into five categories, namely intronic, intergenic, sense, anti-sense (AS), and bidirectional. The recently discovered large lincRNAs (intergenic non-coding RNAs) belongs to the intergenic group which reside in between protein coding region. Unlike lncRNAs, the lincRNAs conserve a similar pattern across various type of species. Thus, a way to distinguished lincRNAs from a sequence is to find these conservatory pattern that marks these actively transcribed genes. For instance, the transcribed region is identified by the trimethylation of lysine 36 of histone H3 (H3K36me3). Meanwhile, the promoter region is identified by the trimethylation of lysine 4 of histone H3 (H3K4me3) [6]. LincRNAs establish cell type-specific epigenetics states which major function is to guide the chromatin-modifying complexes to the appropriate genomic loci.

2.2. MicroRNAs

miRNAs are short single-stranded molecules ranging from 20 to 24 nucleotides which are conserved throughout evolutions. In terms of structure, miRNAs have distinctive hairpin structures. miRNAs are putative translational regulatory gene family [2]. It is belief to actively involved in the post-transciptional process by regulating the expression of half of the genes in a cell [7]. A recent report by Zou [8], suggested that destabilizating a target miRNA greatly reduced the protein levels of a cell [9]. Furthermore, miRNAs also regulate gene expression by activating sequence translation and targeting specific promoters.

2.3. Small Interfering RNAs

A siRNA is of 20-24 nt in length and is commonly described as a linear, perfectly base-paired double stranded RNA. siRNAs were dicovered during transgene-induced silencing in petunia and later moving on to Caenorhabditis elegans. siRNA is a subfamily of RNA interference (RNAi). RNAi was a natural defense mechanic against infection. Yet, siRNA has a similar function to miRNA which facilitate post-transcriptional gene silencing (PTGS). RISC (RNA-induced silencing complex) acts as a medium for siRNA to perform direct silencing with the aids of a Dicer. However, it remained unclear to whether these siRNAs are the main powerhouse for all RNA (RiboNucleicAcid) silencing functions [4].

3. STANDARD EXTRACTED FEATURES

To determine whether a gene is categorized to be non-coding variant, we should first understand the characteristics of feature belonging to non-coding variant itself. Non-coding RNAs (ncRNAs) can be identify through their high occurrences of stop codons and the lack of extensive “Open Reading Frame” (ORF) [10].

3.1. Maximum Coding Subsequence (MCSS)

In the process of identifying non-coding RNAs there will be a huge number of partial-length protein-coding transcripts. The identification of these transcripts has a common limitation whereby there is either a missing start codon or a missing stop codon. Meanwhile, the start codon and stop codon is required in the process of identifying an Open Reading Frame (ORF). This directly affected the CDS prediction. Thus, by putting aside the start codons and stop codons the MCSS is used for the prediction of incomplete CDS across partial-length protein-coding transcripts. MCSS is basically a measure the maximum length of a subsequences derived from a specific sequence which in return provide an estimation of the coding capability of a specific sequence. To achieve this a MCSS score is calculated for each transcript.

Firstly, generate three reading frames with varying starting triplet from the transcript. Next, Kadane’s Algorithm is used to calculate the coding subsequence score for each reading frames. Lastly, compare the coding subsequence score for each transcript with three varying reading frames. The transcript with the maximum value is selected as the MCSS. A pseudocode is illustrated as in Figure 1.

Where hi is the ith hexamer in the reading frame. And the frequency of the hexamer (hi) in a CD and noncoding sequences is denoted by the F(hi) and F’(hi), respectively. Hexamers that is not part of any CDS will have a F(h) sets to zeros. On the other hand, it is possible to find any combination of hexamers in noncoding sequences, so the F’(h) are always larger than zero for noncoding sequences.

```
Initialize:
    max_so_far = 0
    max_ending_here = 0
Loop for each element of the array
    if F(h_i) != 0
        max_ending_here = max_ending_here + log(F(h_i) / F'(h_i))
    else
        max_ending_here = 0
        max_ending_here = max(max_ending_here, 0)
        max_so_far = max(max_ending_here, max_so_far)
Return max_so_far
```

Figure 1. Pseudocode for calculating the Maximum Coding Subsequences (MCSS) Score

3.2. Exon Features

As mentioned earlier there is a huge number of partial length protein. However, the most of previous ncRNAs identification methods are leaning towards complete transcript sequences. The introduction of exon feature is to address this bias issue exists in previous study and ensure the method can perform better in interpreting partial or incomplete transcript such as partial protein-coding transcripts. Exons normally does not contain start codon or stop codon as do partial length protein-coding sequences. Thus, making it an ideal feature for identification of ncRNA. Additionally, based on a study done by Steijger [11], up to 70% of coding exons can be identified using the current transcript assembly methods. An exon with the largest size is selected from each transcript to be the representative exon which is then derived into up to three sub features. The following section will describe the three sub features in details.

The first sub feature is the GC-content of the exon. GC-content as the name implies, is a measurement of the population of G(guanine) and C(cytosine) across all sequence bases. There are normally a high number of G and C bases in the coding regions. The best exon is selected based on the exons with the highest number of GC- content across the same transcript. The second exon feature is the In-frame hexamer frequencies. This sub feature was first used by Claverie [12] to identify coding region. However, it is still being used up until now to distinguish coding transcripts from noncoding transcript. Hexamer frequencies show the relationship between neighboring amino acids in a protein [13]. Lastly, hexamer score distance is calculated in relation to hexamer score is used as an add-on feature to discriminate the coding and noncoding regions. In (1) shows the calculation of Hexamer score distance.

$$Hexamer\ score\ distance\ = \sum (Sm\ -\ Si)\ 3i = 1 \quad (1)$$

The Sm is the maximum hexamer score and Si is the ith hexamer score for the reading frame. Three forward reading frames is generated base on the range of i which is 1, 2 and 3. Each exon in a transcript undergoes the same calculation and the value is used as a parameter estimation to identify non-coding regions.

3.3. Open Reading Frame (ORF)

Although it is proven that ORF based prediction has a lower accuracy for partial-length protein-coding transcripts. ORF still remained as one of the best performing features for full-length protein-coding. Thus, in the matters of classifying ncRNAs from a full-length coding transcript, multiple ORF features can be used simultaneously in order to achieve a better classification performance. For instance, ORF distance, ORF hexamer score, ORF fickett score, ORF coverage and ORF length are all usable ORF features.

Some common ORF feature includes ORF hexamer distance and score, Fickett Score, ORF length, and ORF coverage. ORF hexamer distance and score are like the exon hexamer distance and score. They both utilized the same formula to calculate distance and score. Fickett score is a scoring method used as an alternative to hexamer score. Fickett score was first introduced and used in the identification of protein-coding regions by Fickett. [14] With further experiment, it shows a promising result in the classification of noncoding transcripts and protein-coding transcript [13]. A putative ORF is the longest among all open reading frames and thus making ORF length an essential element in distinguishing the non-coding variant from common transcript. Lastly, ORF coverage is defines by the ORF length over transcript length.

4. DATA/SEQUENCE REPRESENTATION

All algorithm or machine learning model have a sets of representation rules for the input data [15]. This representation is able to aid the execution of a more efficient flow of the algorithm meanwhile reducing execution time and usage of resources [16]. The common representations of these data are in the form of vector or matrix representation. Sequence representation is a way to represents sequences data such as DNA sequence, RNA sequence and protein sequence in a vector representation [17].

4.1. One-Hot Encoding

One hot encoding also sometimes known as one hot vector. One hot vector is a straightforward representation of words or sequences in the form of vector encoding. In [17] a binary vector with a single non-zero value across the dimension. The downside of using this type of encoding is that the dimension grows exponentially to the length of k. For instance, a 3-mer needs a bit of dimension 43 = 64 and for a 4-mer will needs a bit of dimension 44 = 256. Moreover, the distance between any arbitrary pair of one-hot vectors is equidistant, which is impractical for DNA sequences as the GC contents affects the distance between sequences. One-Hot Vector representation of DNA sequence [18].

4.2. Word to Vector Model

Word to Vector model as describe by the name is a way to embed word into a list of vectors or matrix. It is also called the "word2vec" model. In [19], unlike standard word embedding method, word2vec is a neural network which was pre-trained different to produce distributed representation of words. This means that each representation can be of varying size of multiple words forming one representation. This dynamic behavior results in a better natural language processing. This is because the model was able to capture many precise syntactic and semantic word relationships.

5. DEEP LEARNING

5.1. Convolutional Neural Network

Convolutional Neural Network is common in the domain of pattern and image recognition. The networks well in big data and commonly used. The networks do not consist of hidden layer like the other deep neural networks, but instead the networks consist of convolution layer, max pooling layer and fully connected layer. In other word, these networks were a combination of three architecture ideas which ensure some degree of distortion invariance. [20] It was reported that barebone CNN can train a word vectors for text classification tasks with little hyper parameter tuning which focused on sentiment analysis and question and answer classification and achieved a good result on multiple benchmarks. In [21] besides, there are also an upgraded and modified the proposed model to allow the model to run both task-specific and static vectors. In their discussion, it is stated that learning for task specification through fine tuning leads to an improvement in performance. A more advance research of neural network will be focusing on the usage of hardware in conducting the CNN techniques [22]. This includes the usage of graphics cards (GPUs) in 2010 which impressively speed up a normal neural network. However, the sometimes the error rate obtained was higher.

5.2. Deep Learning Framework

Building a deep neural network is not an easy task. Only experts capable to integrate all the computational modelling within the deep neural networks process [23]. Fortunately, thanks to standard inference tools and networks modular structure, several frameworks is introduced in order to helps others to speed up the process in designing and training of neural network models.

One of the most popular deep learning frameworks is TensorFlow introduced by Google recently [24]. This framework provides several enhancements in term of compilation time and graphical visualization. There are many others framework which is suitable in constructing Deep Learning model such as Torch7, Theano, and Caffe. Table 1 shows a comparison of popular deep learning framework.

Table 1. A Comparison of Popular Deep Learning Frameworks

Framework	Core Programming Language	Interfaces from Other Languages	Programming Paradigm	Wrappers
Tensorflow	C++/CUDA	Python	Declarative	Pretty Tensor, Keras, Tensorlite
Caffe	C++/CUDA	Python, Matlab	Imperative	-
Theano	Python (compiled to C++/CUDA)	-	Declarative	Keras, Lasagne or Blocks
Torch7	LuaJIT (with c/CUDA backend)	C	Imperative	-

6. COMPARATIVE ANALYSIS OF COMPUTATIONAL APPROACH AND ITS ALGORITHM

The major type of techniques used to identify the non-coding variant is through computational algorithm or machine learning. This part aims to do a comprehensive comparison of previous research on the tools that have been successfully implemented along with features number and the accuracy. The computational approach and algorithm used for each tool and the advantage and disadvantage of each machine learning algorithm is also tabulated.

6.1. CPAT

Coding Potential Assessment Tool (CPAT) is an alignment-free method which is used to quickly distinguish between coding RNA and noncoding RNA [13]. For simplicity of binary classification, a logistic regression model is implemented inside the tool. The tool considers of features such as open reading frame coverage, open reading frame size, hexamer usage bias and Fickett TESTCODE statistic. The supported sequence format is FASTA and BED. A web application is available for user to predict genes instantly. The source code is implemented in Python and C and is freely available at: http://code.google.com/p/cpat/.

6.2. lncScore

In the research done by Kai Wang and his team in 2016, a logistic regression model was used. This is a study that has provided almost a complete view on coding and non-coding variant identification. The outcome of the research is a tool called lncScore using 11 carefully selected features to identify long noncoding RNA [25].

6.3. CNCI

Coding-Non-Coding Index (CNCI) is a tool that utilized a support vector machine. It is also used to differentiate protein-coding sequences and non-coding sequences without considering known annotations [26]. The sequences are evaluated in an adjoining nucleotide triplets (ANT)The main highlight of this tools is its dynamic construction of usage frequency matrix on the ANT. The matrix is used as the bases for the calculation of features such as the length- percentage, codeon-bias, score-distance, length and S-score of the target transcript.

6.4. PLEK

Based on the study done by Aimin Lin and his team in 2014, a support vector machine (SVM) with improved k-mer scheme was proposed. They name the study as PLEK [27]. The k-mer scheme is a list of specific combination fo nucleotide forming a string with the length of k Thus, the name as k-mer pattern list. A nucleotide can be either one of the alphabet from A, C, G and T. Assuming k is equal to 1 to 4, will equates to 4 + 16 + 64 + 256 = 340 patterns: 4 one-mer patterns, 16 two-mer patterns, 64 three-mer patterns and 256 four-mer patterns. These 340 calibrated k-mer usage frequencies are calculated for each transcript and fed into the SVM algorithm. Besides that, this new strategy made the identification process much faster, yet in exchange of higher requirement of processing power.

6.5. lncRNA-MFDL

lncRNA-MFDL utilize a deep learning model to classify coding and noncoding RNA [28]. It introduces deep stacking network (DSNs) which involves stacking of multiple shallow network. The model is comprised of three stages which are feature extraction, feature fusion and pattern classification. The first stage is described as plenty of shallow networks categorizing into modules, namely ORF descriptor, k-mer descriptor, SS descriptor and MLCDS descriptor. The second stage fuse all the shallow networks in stage one into a single transcript representation which is another shallow network. And for the last stage, pattern classification, the transcript representation from previous stage will be classify. Both the input and output of the classifier will be in vector representation.

6.6. Justification

The review shows that there are plenty of tools used for non-coding variant identification. Yet the tools rely heavily on inconsistent features which greatly affect the performance of the algorithm. Table 2 shows the comparative analysis of some of the approaches that have been performed in previous studies. Based on the summarized of noncoding RNA tools in Table 2. The type of features plays some crucial role in better identification accuracy for instance the lncScore achieved a better accuracy of 96.46% using logistic regression as compared to CPAT with an accuracy of 94.65 using a different type of features. The lncScore have a more diverse categories of features evaluating the Exon features, MCSS features and ORF features as compared to CPAT which focus mostly on the ORF features. The increase in the number of features for algorithm training does not guarantee a better accuracy as the PLEK tools has a total of 1364 features yet it

only achieves an accuracy of 83.67%. This is far left behind as compared to CNCI which utilizing the same method yet achieved an accuracy of 93.4%. Meanwhile, the last tools lncRNA-MFDL has the best accuracy of 97.1% as compared to others. This could mainly due to the underlying algorithm which is a deep learning model called Deep Stacking Network as defined by the author. It had been proven that a deep learning neural network model can achieved a better result for prediction and classification problems.

From previous research, there is yet any notable tools that utilizing convolutional neural network as the base's framework. Despite the facts that convolutional neural network is able to perform feature extraction automatically and achieved a better accuracy in prediction problem as a deep learning model [29]. With the success of deep learning neural network bases model implemented as per the lncRNA-MFDL, Convolutional Neural Network yet another deep neural network has a great potential in non-coding variant discovery as it does not rely on the current discovered feature yet was able to learn the type of feature in abstracted form which is best suit for a certain variant identification.

Table 2. Non-Coding RNA Tools Comparison

Tools Name	No of features	Algorithm	Advantage	Disadvantage
CPAT	4	Logistic Regression	- Simple and easy to implement - Able to predict accurate results for most classification problems	- The precision of classifier decreases as the size of data decrease. -Need large number of sample and data
lncScore	11	Logistic Regression		
CNCI	5	Support Vector Machine	- High accuracy	- Required speed and size of processing in both training and testing data. - More complex - Need extensive memory requirements for classification
PLEK	1364	Support Vector Machine	- Organized and functioning well even the data is not linearly separable in the feature space	
lncRNA-MFDL	5	Deep Learning method-Deep Stacking Networks	- High ac- More representative and discriminative for scene categorization. - Accelerated by general-purpose graphic units (GPUs) - High optimization performance - More efficient algorithm and class-independent algorithms over class-independent algorithms over class-specific approaches.	- The nature of the learned representations remains unclear when implemented - Cannot provide details about objects and their layouts in images - Need large memory of GPU if process large amount of data.

7. CONCLUSION

In conclusion, this paper shows a brief overview on the key aspect of non-coding variant identification. From the general information of classification approaches and the implementation of deep learning principles using deep learning framework in genomic. Not to mention, the significance, and current trend of genomic study also being stated in this paper.

REFERENCES

[1] Yi, X., Zhang, Z., Ling, Y., Xu, W., & Su, Z., "PNRD: A plant non-coding RNA database". *Nucleic Acids Research*, vol. 43, no. D1, pp. D982–D989, 2015.
[2] Eddy, S. R., & Hughes, H., "Non-Coding Rna Genes and the modern RNA world," *Genetics*, 2(December), pp. 919-929, 2001.
[3] Kaikkonen, M. U., Lam, M. T. Y., & Glass, C. K., "Non-coding RNAs as regulators of gene expression and epigenetics," *Cardiovascular Research*, vol. 90, no. 3, pp. 430-440, 2011.
[4] Costa, F. F., "Non-coding RNAs: Lost in translation?," *Gene*, 2007.
[5] Volders, P. J., Helsens, K., Wang, X., Menten, B., Martens, L., Gevaert, K., Mestdagh, P., "LNCipedia: A database for annotated human lncRNA transcript sequences and structures." *Nucleic Acids Research*, vol. 41 no. D1, pp. 246-251, 2013.
[6] Lv, J., Huang, Z., Liu, H., Liu, H., Cui, W., Li, B., Wu, Q., "Identification and characterization of long intergenic non-coding RNAs related to mouse liver development," *Molecular Genetics and Genomics*, pp. 1225-1235, 2014.
[7] Akman, H. B., & Bensan, Herson, A. E., "Noncoding RNAs and cancer.," *Turkish Journal of Biology*, vol. 38, pp. 817-828, 2014.
[8] Zou, Q., Hu, Q., Guo, M., & Wang, G., "Sequence analysis HAlign : Fast multiple similar DNA / RNA sequence alignment based on the centre star strategy, " pp. 2475-2481, 31, March, 2015.

[9] Lv, J., Liu, H., Yu, S., Liu, H., Cui, W., Gao, Y., Wu, Q., "Identification of 4438 novel lincRNAs involved in mouse pre-implantation embryonic development," *Molecular Genetics and Genomics*, vol. 290, no. 2, pp. 685-697, 2015.
[10] Costa, F. F., "Non-coding RNAs: Meet thy masters.," *BioEssays*, 2010.
[11] Steijger, T., Abril, J. F., Engström, P. G., Kokocinski, F., Consortium, T. R., Hubbard, T. J., Bertone, P. "Assessment of transcript reconstruction methods for RNA-seq, " vol. 10, no. 12, 2013.
[12] Claverie, J., "Computational methods for the identification of genes in vertebrate genomic sequences", vol. 6, no. 10, pp. 1735-1744., 1997.
[13] Wang, L., Park, H. J., Dasari, S., Wang, S., Kocher, J., & Li, W., "CPAT : Coding-Potential Assessment Tool using an alignment-free logistic regression model," vol. 41, no. 6 , pp. 1-7, 2013.
[14] Fickett, J. W., *Nucleic Acids Research*, vol. 10, no. 17, 1982.
[15] Louridas, P., & Ebert, C., "Machine Learning". *IEEE Software*, vol. 33, no. 5, pp. 110-115, 2016.
[16] Bengio, Y., "Learning Deep Architectures for AI," *Foundations and Trends® in Machine Learning*, vol. 2, 2009.
[17] Ng, P., "dna2vec: Consistent vector representations of variable-length k-mers", pp. 1-10, 2017, Available: Arxiv.org, http://arxiv.org/abs/1701.06279
[18] Nguyen, N. G., Tran, V. A., Ngo, D. L., Phan, D., Lumbanraja, F. R., Faisal, M. R., Satou, K. "DNA Sequence Classification by Convolutional Neural Network," *Journal of Biomedical Science and Engineering*, vol. 09, no. 05, pp. 280-286, 2016.
[19] Mikolov, T., Chen, K., Corrado, G., & Dean, J. "5021-Distributed-Representations-of-Words-and-Phrases-and-Their-Compositionality", pp. 1-9, 2013.
[20] Bai, S., "Growing random forest on deep convolutional neural networks for scene categorization," *Expert Systems with Applications*, vol. 71, pp. 279-287, 2017.
[21] Kim, Y., "Convolutional Neural Networks for Sentence Classification," 2014.
[22] Cireşan, D. C., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J., "*Flexible, high performance convolutional neural networks for image classification*," IJCAI International Joint Conference on Artificial Intelligence, pp. 1237-1242, 2011.
[23] Abadi, X. M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Zheng, X., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," *None*, vol. 1, no. 212, p. 19, 2015.
[24] Rampasek, L., & Goldenberg, A., "TensorFlow: Biology's Gateway to Deep Learning? " *Cell Systems*, vol. 2, no. 1, pp. 12-14., 2016.
[25] Zhao, J., Song, X., & Wang, K., "lncScore: alignment-free identification of long noncoding RNA from assembled novel transcripts. " *Sci Rep*, vol. 6, no. 34838, 2016.
[26] Sun, L., Luo, H., Bu, D., Zhao, G., Yu, K., Zhang, C., Zhao, Y., "Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts," vol. 41, no. 17, 2013.
[27] Li, A., Zhang, J., & Zhou, Z., "PLEK : a tool for predicting long non-coding RNAs and messenger RNAs based on an improved k- mer scheme," pp. 1-10, 2014.
[28] Fan, X.-N., & Zhang, S.-W., "lncRNA-MFDL: identification of human long non-coding RNAs by fusing multiple features and using deep learning," *Mol. BioSyst.*, vol. 11, no. 3, pp. 892-897, 2015.
[29] Shustanov, A., & Yakimov, P. "*CNN Design for Real-Time Traffic Sign Recognition*." Procedia Engineering, vol. 201, pp. 718-725, 2017.

BIOGRAPHIES OF AUTHORS

Kuan Xin Lee received his B.Sc. degree (1st class Hons.) in computer science from the University of Teknology Malaysia in 2018. He is currently pursuing the Ph.D degree in the Department of computer Science, University of Technology Malaysia. His research interests include data mining, machine learning and bioinformatics.

Dr Afnizanfaizal Abdullah received his B.Sc. degree in computer science from the University of Teknology Malaysia in 2007. He then received Master of Science (Computer Science) 2009 and Doctor of Philosophy (Computer Science) 2013 from University of Technology Malaysia. He is senior lecturer at the School of Computing that specializing in artificial intelligence techniques for analyzing biological data. His research interests are in the designing of machine learning algorithms for healthcare applications in the cloud environments. In 2015, he have co-founded Synthetic Biology Research Group to drive innovation in research and development of healthcare, biotechnology, and environment areas through computing and engineering.