

## Research Article

# Deep Learning for the Classification of Genomic Signals

**J. Alejandro Morales** <sup>1</sup>, **Román Saldaña** <sup>1</sup>, **Manuel H. Santana-Castolo** <sup>1</sup>,  
**Carlos E. Torres-Cerna** <sup>1</sup>, **Ernesto Borrayo** <sup>2</sup>, **Adriana P. Mendizabal-Ruiz** <sup>3</sup>,  
**Hugo A. Vélez-Pérez** <sup>1</sup> and **Gerardo Mendizabal-Ruiz** <sup>1</sup>

<sup>1</sup>*Departamento de Ciencias Computacionales, Universidad de Guadalajara, Guadalajara, Mexico*

<sup>2</sup>*Departamento de Electrónica, Universidad de Guadalajara, Guadalajara, Mexico*

<sup>3</sup>*Departamento de Farmacobiología, Universidad de Guadalajara, Guadalajara, Mexico*

Correspondence should be addressed to Gerardo Mendizabal-Ruiz; [gerardo.mendizabal@academicos.udg.mx](mailto:gerardo.mendizabal@academicos.udg.mx)

Received 9 July 2019; Revised 22 July 2019; Accepted 6 November 2019; Published 5 May 2020

Guest Editor: Gonzalo Pajares

Copyright © 2020 J. Alejandro Morales et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Genomic signal processing (GSP) is based on the use of digital signal processing methods for the analysis of genomic data. Convolutional neural networks (CNN) are the state-of-the-art machine learning classifiers that have been widely applied to solve complex problems successfully. In this paper, we present a deep learning architecture and a method for the classification of three different functional genome types: coding regions (CDS), long noncoding regions (LNC), and pseudogenes (PSD) in genomic data, based on the use of GSP methods to convert the nucleotide sequence into a graphical representation of the information contained in it. The obtained accuracy scores of 83% and 84% when classifying between CDS vs. LNC and CDS vs. PSD, respectively, indicate the feasibility of employing this methodology for the classification of these types of sequences. The model was not able to differentiate from PSD and LNC. Our results indicate the feasibility of employing CNN with GSP for the classification of these types of DNA data.

## 1. Introduction

In molecular biology, the genome is the complete genetic information of an organism. The genome, in the form of DNA, is a relatively large molecule strung by a series of nucleotides arranged in particular manners for every organism, granting the sense that such arrangements and patterns are the information that determines it. A genome is capable of defining an organism due to numerous processes that involve reading small DNA fragments of the genome to enable the cells to compile new molecules, such as RNA or proteins. The resulting molecules will exert various functions in the cell, from transforming their surroundings into energy and more cells, sensing the environment, moving, and self-regulating all of the abovementioned procedures. This last function holds particular relevance, for it gives the system flexibility, adaptability, and robustness while approaching near-optimal performance at steady state. At the same time,

regulation complexity is directly related to the organization state of an organism. For example, mammals have greater regulation complexity than insects, and both, than bacteria.

The organisms display a myriad of genomic functions. The most widely known genomic function is the coding regions (CDS). These are sequences that, after transcription and maturation into mRNA, will encode for proteins and therefore share very few characteristics: codon modulus and some motifs for particular protein families.

A couple of decades ago, it was still believed that mRNAs were the only transcription products worth studying. However, recent studies have proven the opposite, uncovering several regulatory transcription functions. Some of these functions are performed by long noncoding RNAs (LNC) and pseudogenes (PSD). Long noncoding regions are highly heterogeneous [1] transcribed regions >200 bp that exhibit similar characteristics as mRNA (e.g., polyadenylation and splicing [2]), yet they have mainly been

shown to influence transcription rate by *cis*- and *trans*-acting over different genes [3]. Furthermore, nuclear structural functions have been attributed, with the ability of LNC to scaffold [4] and guide DNA localization [5], and in the cytoplasm, LNC interacts with other RNAs to regulate their decay, CDS translation rate, and protein signaling [6]. Many of these functions require that (at least) part of the LNC region to be homologous or complementary to its CDS target.

Other DNA regions similar to LNC are PSD. These are regions with high homology to CDS that have lost their capacity to translate and code for proteins [7]. Some of them may transcribe, and such regions will usually interfere with their homologous CDS in the RNA regulation network, altering the latter's decay or translation rate [8].

Given the fact that this is a complex, self-encoded, self-regulating system, a great deal of effort has been put in understanding whether there exist information patterns governing each functional type, and if so, whether can we identify them and differentiate from other functional types. To answer these questions, several tools have been used; among them, digital signal processing (DSP) provides a set of novel and useful tools for solving highly relevant problems.

Genomic signal processing (GSP) is a field of bioinformatics that is based on the use of digital signal processing methods for the analysis of genomic data [9]. GSP provides a set of novel and useful tools for processing and understanding of the vast information that is currently available in sequenced genomes from many living organisms [10]. GSP methods rely on a procedure that consists of mapping the data originally in the form of a string of characters (e.g., A, T, C, and G, for DNA) to a numeric representation that can be processed using numeric algorithms deployed on digital systems such as computers or digital signal processors. Many mapping methods have been proposed [11–13] for different applications that include the classification of exon and intron sequences [14], the comparison of alignment-free genetic distances [15], the clustering analysis of species [16], the classification of viruses [17], the classification of gene sequences as diseased or nondiseased state [18], and fast genome classification at different taxonomic levels [19]. Through the use of GSP, it is possible to generate alternative representations for the information contained in the DNA sequences. However, due to the high complexity of this data, the analysis of such information is a challenging task.

Deep learning is an area of computer science that attempts to replicate the way that human brains learn by creating and training artificial neural networks. Deep learning has rapidly become the most successful of the machine learning methodologies for solving classification tasks. The success of deep learning in comparison with other approaches is because deep learning allows the generation of computational models that are composed of multiple processing layers that are capable of learning representations of data with multiple levels of abstraction [20].

In particular, the convolutional neural networks (CNN) have become the leading deep learning architecture for most image recognition, classification, and detection tasks [21]. While there exist many variants of CNN, in general, they consist of many convolutional and pooling layers stacked on top of each other followed by dense fully connected layers. The convolutional layers accomplish the function of extracting relevant feature representations from the input image data. On the other hand, the pooling layers reduce the spatial resolution of the convoluted images to achieve spatial invariance and reducing the complexity of the network. Finally, the fully connected layers perform the classification of those features extracted by the convolution modules. CNN have been widely applied to successfully solve many problems including the classification of objects in images [22], human actions recognition [23], large-scale video classification [24], face detection [25], image style transfer [26], volumetric image segmentation [27], and several medical image analysis applications [28]. Moreover, CNN have been successfully applied to detect faulty components based on the use of signal processing techniques to feed the network [29–31].

In this paper, we present a methodology based on GSP and deep learning for the classification of three distinct functional genome types: coding regions, long noncoding regions, and pseudogenes. The proposed methodology consists of converting the DNA nucleotide sequence into a graphical representation of the information contained in it (i.e., spectrogram) and the implementation of a CNN to create a model capable of discriminating between coding and long noncoding regions and coding regions and pseudogenes. We obtained an accuracy scores of 83% and 84% when classifying between CDS vs. LNC and CDS vs. PSD, respectively. Our results indicate the feasibility of employing this methodology for the classification of these types of sequences.

## 2. Materials and Methods

Given a DNA sequence  $S$  of length  $L$ , where  $S[j] = A, T, C, G$ , this is mapped into a numeric representation  $\hat{S}$  using the Voss representation [32] which consists of generating four vectors  $\hat{S}_i$  with  $i \in [A, T, G, C]$ , where

$$\begin{aligned}\hat{S}_A[j] &= \begin{cases} 1, & \text{if } S[j] = A, \\ 0, & \text{otherwise,} \end{cases} \\ \hat{S}_G[j] &= \begin{cases} 1, & \text{if } S[j] = G, \\ 0, & \text{otherwise,} \end{cases} \\ \hat{S}_C[j] &= \begin{cases} 1, & \text{if } S[j] = C, \\ 0, & \text{otherwise,} \end{cases} \\ \hat{S}_T[j] &= \begin{cases} 1, & \text{if } S[j] = T, \\ 0, & \text{otherwise.} \end{cases}\end{aligned}\tag{1}$$

For example, for the sequence  $S = \text{AACTGT}$  the corresponding Voss vectors are  $\hat{S}_A = [110000]$ ,  $\hat{S}_C = [001000]$ ,  $\hat{S}_T = [000101]$ , and  $\hat{S}_G = [000010]$ .

Since each of the vectors  $\hat{S}_i$  can be seen as a digital signal that represents the patterns of occurrence of its corresponding nucleotide type, it is possible to perform a frequency analysis of each of those signals by estimating the power spectra of the signal by using a short-time Fourier transform (STFT) analysis [33].

STFT consists of dividing the input signal into shorter segments of equal length and then computing the Fourier transform on each of the short segments. The resulting data is concatenated to generate a matrix with the frequency component's strength with respect to the length of the original signal. This matrix is commonly converted into a color image by applying a color palette which represents the intensity of each frequency component.

In this work, we generated a single signal vector  $\hat{Z}$  by concatenating the four signals  $\hat{S}_i$ . Then, we performed STFT analysis on  $\hat{Z}$  using a sliding window of size  $W$  with  $O$  overlapping points between consecutive sliding windows. As a result, we obtained the matrix  $M$  which is converted to an image that represents the repetition patterns of the nucleotides (i.e., spectrogram) by performing a normalization of the values of  $M$  which then are converted to indexes of a "Jet" color palette (e.g., Figure 1).

To classify the generated spectrograms, we created a 18-layer CNN with an architecture based on the use of four stacked inception type A modules with a structure similar to the one proposed in the Inception V3 model [34].

This model allowed us to capture the power of the original inception architecture without its high complexity, which is unnecessary considering that we are dealing with a binary classification problem instead of the 1,000 categories of the ImageNet database [35] for what this model was created. Figure 2 depicts a diagram of the proposed CNN architecture. In the first layers, we employed seven convolutional layers with max pooling that allowed to extract features from the spectrograms while reducing the size of the data to analyze. Then, we stacked four Inception V3 modules that allowed the network to learn the complex patterns in the spectrogram features at different levels of depth. The two first inception layers have a depth of 256 convolution filters while the second two reduce the dimensionality of the features to 128 filters. Finally, we employed three layers of fully-connected neurons to classify the features extracted in previous layers. Finally, we added an output layer with two neurons. In total, the proposed network consists of 22,179,500 trainable parameters. Table 1 lists the details of the proposed network architecture.

The network was trained using a stochastic gradient descent optimizer with a learning rate of  $1 \times 10^{-5}$ , support for the momentum of 0.9, and a learning rate decay of (01/100). The loss function used for training the model is the binary cross entropy. Additionally, we employed a reduction of the learning rate on a plateau whenever the reduction in the validation loss was less than  $1 \times 10^{-4}$  for more than five consecutive epochs.

### 3. Results

We collected DNA sequences from the NCBI FTP database ([ftp.ncbi.nlm.nih.gov/genomes/refseq/assembly\\_summary\\_refseq.txt](ftp.ncbi.nlm.nih.gov/genomes/refseq/assembly_summary_refseq.txt)) corresponding to CDS, LNC, and PSD of not closely related species with a very high frequency of fragments of at least 1,000 bp. From these, we randomly selected 15,000 fragments of 1,000 bp for each type, including all species listed.

We generated a set **A** consisting of 10,000 spectrograms of CDS and LNC sequences (5,000 for each type) employing Python's Matplotlib library using windows of size  $W = 512$  with an overlap of  $O = 500$  samples. Then, the spectrogram images were resized to  $224 \times 224$  through bilinear interpolation. The CNN model was trained during 50 epochs with a batch size of 50 images. Eighty percent of the set **A** was used for training the model on each epoch, and the rest was used for validation during the training.

Figure 3 depicts the loss function value and the model's accuracy with respect to the epoch. Note that the model learns rapidly the image spectrogram patterns during the first epochs for each class, obtaining an accuracy score of around 80 percent. However, the model performance on the validation set oscillates during more than twenty-five epochs to finally converge to a stable value. A set of 20,000 never seen DNA spectrograms were generated for testing the model (10,000 for each class). The weights that obtained the best loss function value during the training were employed for the classification of this dataset. The accuracy obtained using the proposed model was 0.83 with a F1 score of 0.84. Table 2 lists the confusion matrix scores for each classification result type.

Next, we generated a second set **B** consisting of the previously employed 5,000 CDS samples and 5,000 more samples of PSD sequences using the same parameters employed for the previous CDS vs. LNC model. Figure 4 depicts the loss function value and the model's accuracy with respect to the epoch. Note that the model reached a high score in the validation during the first epochs, which is maintained during the whole training process with a convergence after the 50 epochs. One more time, we tested the model with a set of 20,000 never seen DNA spectrograms (10,000 for each class). In this experiment, we obtained an accuracy score of 0.84 with an F1 score of 0.85. Table 3 lists the confusion matrix scores. Note that the proposed model was able to correctly classify the majority of the sequences in the set, with a slightly higher score than in the case of CDS vs. LNC.

Finally, we generated a third set **C** consisting of the 5,000 LNC and the 5,000 PSD sequences previously employed in sets **A** and **B**, respectively. Figure 5 depicts the loss function value and the model's accuracy with respect to the epoch. Note that the model was not able to make the validation curve to converge after the 50 epochs. We can note that the model is overfitting since it has high accuracy on the training set.

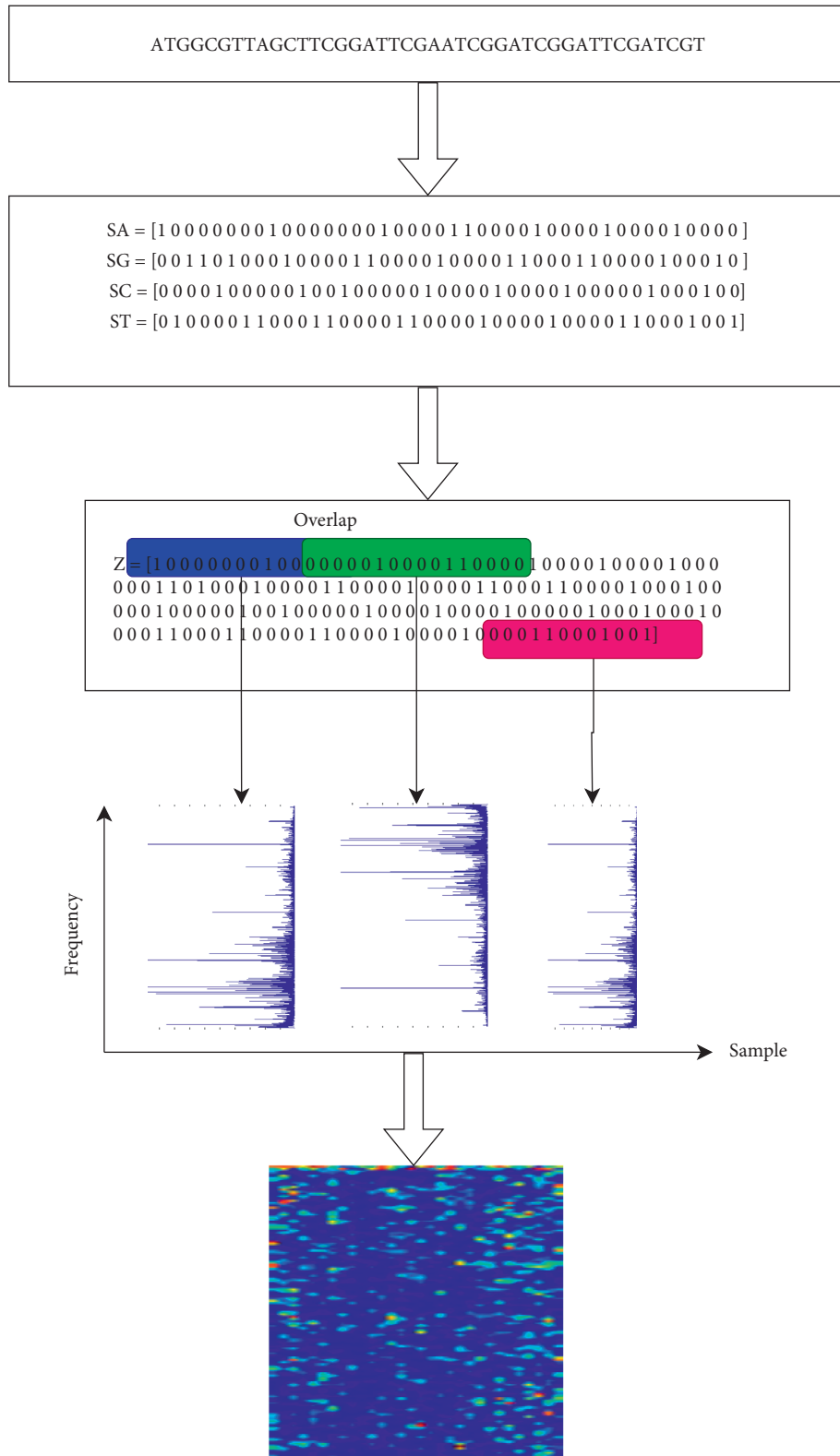


FIGURE 1: Diagram of the DNA sequence for spectrogram conversion procedure.

#### 4. Discussion

In recent years, the study of LNC has grown in number and detail. Previous studies have shown that as far as 68% of the

human transcriptome is classified as LNC [36]. Ever since it has been revealed that around 94% of human DNA transcribes into RNA [37], these functions have been extensively studied, specially LNC sequences in humans with useful

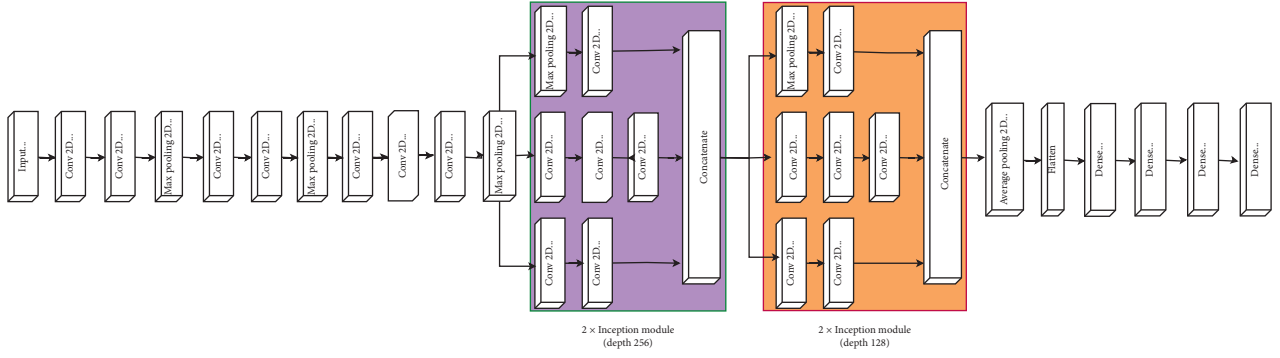


FIGURE 2: Diagram of the proposed CNN architecture.

TABLE 1: Description of the layers for the proposed CNN architecture.

Type of layer	Patch size	Output size	Number of Parameters
Input	$3 \times 3$	$224 \times 224 \times 3$	0
Convolution	$3 \times 3$	$222 \times 222 \times 32$	896
Convolution	$3 \times 3$	$220 \times 220 \times 64$	18,496
Max pooling	$3 \times 3$	$109 \times 109 \times 64$	0
Convolution	$1 \times 1$	$109 \times 109 \times 80$	5,200
Convolution	$3 \times 3$	$107 \times 107 \times 192$	138,432
Max pooling	$3 \times 3$	$53 \times 53 \times 192$	0
Convolution	$3 \times 3$	$51 \times 51 \times 64$	110,656
Convolution	$3 \times 3$	$49 \times 49 \times 90$	51,930
Convolution	$3 \times 3$	$47 \times 47 \times 192$	155,712
Max pooling	$3 \times 3$	$23 \times 23 \times 192$	1
Inception	$3 \times 3$	$23 \times 23 \times 768$	1'918,464
Inception	$3 \times 3$	$23 \times 23 \times 768$	2'360,832
Inception	$3 \times 3$	$23 \times 23 \times 384$	590,464
Inception	$3 \times 3$	$23 \times 23 \times 128$	443,008
Average pooling	$3 \times 3$	$9 \times 9 \times 48$	0
Fully connected		$1 \times 1 \times 512$	15'925,760
Fully connected		$1 \times 1 \times 256$	131,328
Fully connected		$1 \times 1 \times 128$	32,896
Softmax		$1 \times 1 \times 2$	258

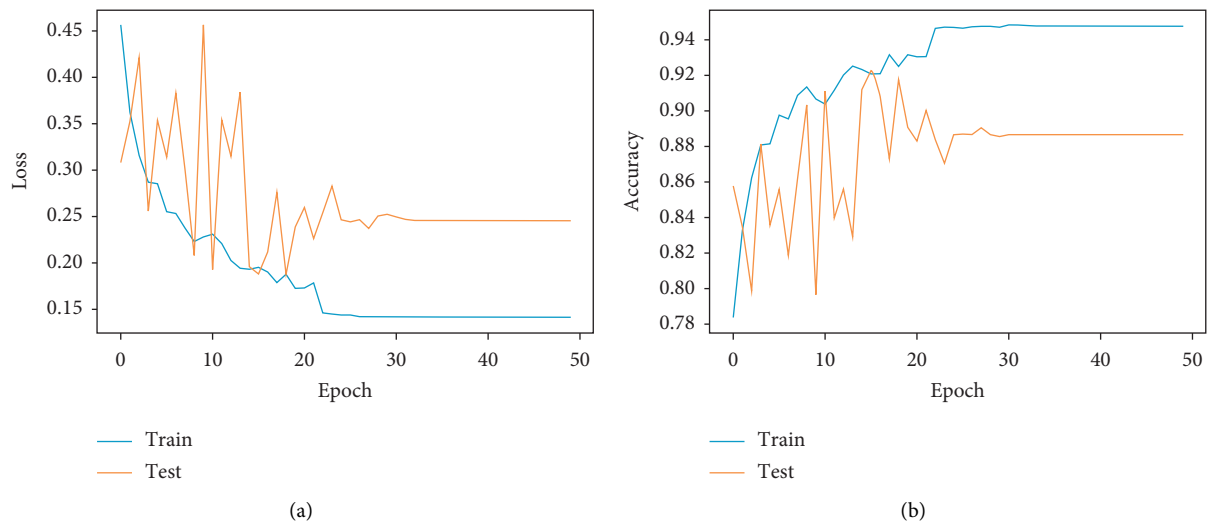


FIGURE 3: Training curves for datasets CDS and LNC during 50 epochs. (a) Model loss. (b) Model accuracy.

TABLE 2: Confusion matrix obtained on the test dataset for testing CDS and LNC sequences.

	Predicted CDS	Predicted LNC
True CDS	9,058	942
True LNC	2,358	7,642

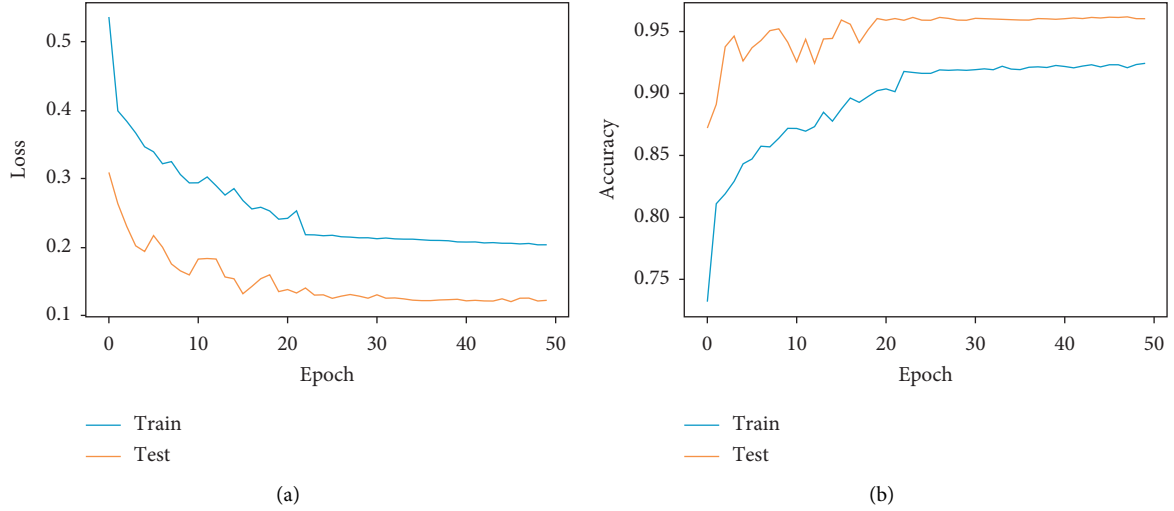


FIGURE 4: Training curves for datasets CDS and PSD during 50 epochs. (a) Model loss. (b) Model accuracy.

TABLE 3: Confusion matrix obtained on the test dataset for testing CDS and PSD sequences.

	Predicted CDS	Predicted PSD
True CDS	8,804	1,196
True PSD	1,835	8,165

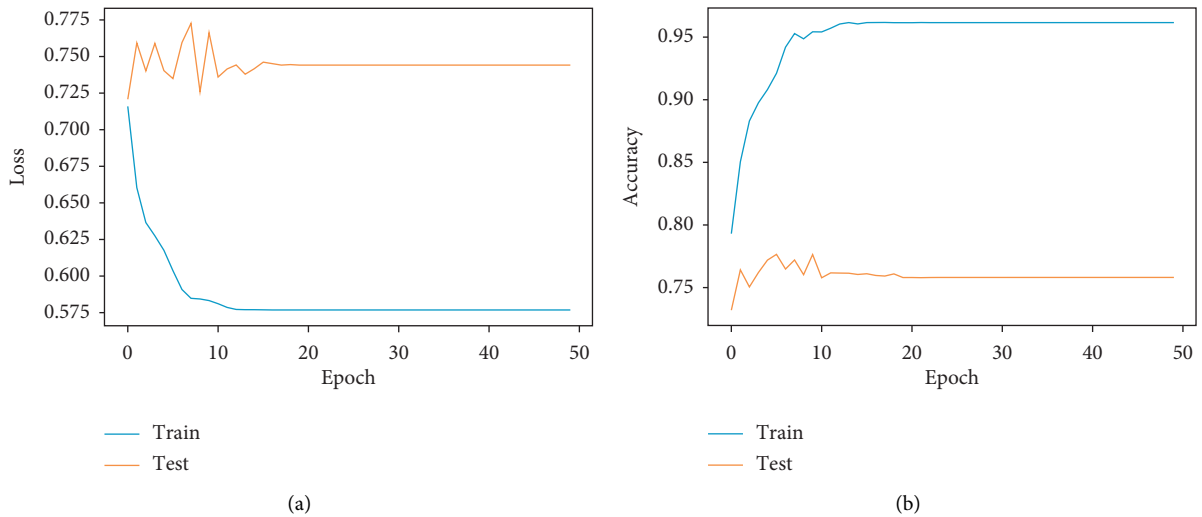


FIGURE 5: Training curves for datasets LNC and PSD during 100 epochs. (a) Model loss. (b) Model accuracy.

applications. For example, the presence, frequency, and types of LNC and PSD in various human tissues are being developed as markers for cancer diagnosis and prognostics [8, 38].

According to Kopp and Mendell [3], LNC sequences do not perform sequence-specific functions which would

explain their high heterogeneity. However, many of the several functions that have been attributed to LNC-*cis-trans*-regulation, miRNA sponge, seem related, partially at least, to sequences. The same observation could also explain PSD variability as they are currently considered as subfunctions of LNC. In this work, we attempted to distinguish between

these types of sequences by analyzing the frequency patterns using a deep learning classifier. However, from the obtained results during model training, we cannot verify that these differences exist as in the cases of CDS vs. LNC or CDS vs. PSD, and therefore, it is feasible that the intrinsic sequence heterogeneity has no particular frequencies.

When comparing CDS vs. PSD, we obtained the highest accuracy corroborating the spectral and functional divergence between both sequences. These sequences are the result of genomic homology analysis and annotation against CDS, but with premature stop codons and other typical regulatory sequences. Further transcriptome analysis corroborates either their low expression profile or differential expression against its CDS counterparts. Some of the best-described PSD functions include miRNA sequestering and antisense effects to regulate mRNA decay and translation rate. Being part of other biological functions (i.e., CDS produce proteins while PSD regulates mRNA), even when many of the PSD depends on their CDS, they are under different evolutionary pressures, and thus, they will mutate and diverge at their own rate, resulting in high PSD variability that the proposed deep learning model was able to distinguish from CDS.

Other studies have used deep learning methods to identify LNC. Most of them have focused on human lncRNA identification [39, 40] and have used full or *k-mer* sequence data and other features, such as function annotation and dimensional representations [41], secondary structure and ORFs [42, 43], or nuclear location [44]. However, to the best of our knowledge, this is the first deep learning model that is capable of identifying LNC using frequency information solely.

The main advantage of employing a deep learning approach in comparison with other machine learning classifiers rely on the ability of the convolutional neural network to automatically extract the best features that allow achieving a good performance in the classification of the selected type of sequences. For this reason, it is very common that researchers employ existing successful architectures such as Inception V3 or Inception-Resnet V2 to perform classification tasks. However, many of these networks were conceived to perform classification on data with a large number of labels. Therefore, these architectures are very deep (i.e., have many layers) which translates into a more considerable complexity of the network and a more significant number of parameters to optimize and, therefore, larger training times. In this work, we built our convolutional deep learning classifier inspired by the inception modules which have proved to be very good to extract features from the image data. In our experiments, the four inception modules, along with the proposed convolutional and fully connected layers, allow us to achieve a good performance in the classification task using the computed sequences spectrograms.

While our results indicate a performance of around 80% accuracy for both classifiers that may be very useful for many applications regarding automatic detection of coding regions in DNA or the understanding of the relationships between LNC, PSD, and CDS sequences, we believe that this relative low accuracy is due to high similarity in the patterns

of some sequences that are difficult to differentiate for the proposed network architecture. We believe that this value could be increased by others employing other deep learning architectures; however, that is a subject of future work. Additionally, it may be possible to generate deep learning models for the classification of other interesting functional genome sequences such as miRNA or snRNA. However, the dataset's imbalance related to human genome overrepresentation remains an issue that will fade as nonhuman data becomes available.

## 5. Conclusion

We have presented a methodology for the classification of coding regions, long noncoding regions, and pseudogenes based on the use of GSP for converting the DNA nucleotide sequence into a graphical representation of the information contained in the DNA sequences which are later classified using a convolutional deep learning model. The obtained accuracy scores of 83% and 84% when classifying between CDS vs. LNC and CDS vs. PSD, respectively, indicate the feasibility of employing this methodology for the classification of these types of sequences.

## Data Availability

The datasets and CNN model and weights used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

The authors thank the Science and Technology National Council (CONACYT) for scholarship support of RS, MHSC, and CETC (nos. 483669, 429399, and 429397), respectively. This study was supported for CONACYT Grant Basic-Sciences (no. 252465).

## References

- [1] L. L. Chen and G. G. Carmichael, "Long noncoding RNAs in mammalian cells: what, where, and why?" *Wiley Interdisciplinary Reviews: RNA*, vol. 1, no. 1, pp. 2–21, 2010.
- [2] K. Kashi, L. Henderson, A. Bonetti, and P. Carninci, "Discovery and functional analysis of lncRNAs: methodologies to investigate an uncharacterized transcriptome," *Biochimica et Biophysica Acta (BBA) - Gene Regulatory Mechanisms*, vol. 1859, no. 1, pp. 3–15, 2016.
- [3] F. Kopp and J. T. Mendell, "Functional classification and experimental dissection of long noncoding RNAs," *Cell*, vol. 172, no. 3, pp. 393–407, 2018.
- [4] J. M. Calabrese and T. Magnuson, *Molecular Biology of Long Non-coding RNAs*, Springer, New York, NY, USA, 2013.
- [5] P. Han and C.-P. Chang, "Long non-coding RNA and chromatin remodeling," *RNA Biology*, vol. 12, no. 10, pp. 1094–1098, 2015.

- [6] K. Zhang, Z.-M. Shi, Y.-N. Chang, Z.-M. Hu, H.-X. Qi, and W. Hong, "The ways of action of long non-coding RNAs in cytoplasm and nucleus," *Gene*, vol. 547, no. 1, pp. 1–9, 2014.
- [7] C. Glenfield and A. McLysaght, "Pseudogenes provide evolutionary evidence for the competitive endogenous RNA hypothesis," *Molecular Biology and Evolution*, vol. 35, pp. 2886–2899, 2018.
- [8] X. Hu, L. Yang, and Y.-Y. Mo, "Role of pseudogenes in tumorigenesis," *Cancers*, vol. 10, no. 8, p. 256, 2018.
- [9] I. Shmulevich and E. R. Dougherty, *Genomic Signal Processing*, vol. 50, Princeton University Press, Princeton, NJ, USA, 2014.
- [10] D. Anastassiou, "Genomic signal processing," *IEEE Signal Processing Magazine*, vol. 18, no. 4, pp. 8–20, 2001.
- [11] G. Mendizabal-Ruiz, I. Román-Godínez, S. Torres-Ramos, R. A. Salido-Ruiz, and J. A. Morales, "On dna numerical representations for genomic similarity computation," *PLoS One*, vol. 12, Article ID e0173288, 2017.
- [12] C. Yin, "Encoding dna sequences by integer chaos game representation," 2017, <https://arxiv.org/abs/1712.04546>.
- [13] N. Yu, Z. Li, and Z. Yu, "Survey on encoding schemes for genomic data representation and feature learning—from signal processing to machine learning," *Big Data Mining and Analytics*, vol. 1, pp. 191–210, 2018.
- [14] M. Abo-Zahhad, S. M. Ahmed, and S. A. Abd-Elrahman, "Genomic analysis and classification of exon and intron sequences using dna numerical mapping techniques," *International Journal of Information Technology and Computer Science*, vol. 4, no. 8, pp. 22–36, 2012.
- [15] E. Borrayo, G. Mendizabal-Ruiz, H. Vélez-Pérez, R. Romo-Vázquez, A. P. Mendizabal, and A. Morales, "Genomic signal processing methods for computation of alignment-free distances from dna sequences," *PLoS One*, vol. 9, Article ID e110954, 2014.
- [16] G. Mendizabal-Ruiz, I. Román-Godínez, S. Torres-Ramos, R. A. Salido-Ruiz, H. Vélez-Pérez, and J. A. Morales, "Genomic signal processing for dna sequence clustering," *PeerJ*, vol. 6, Article ID e4264, 2018.
- [17] E. Adetiba, O. O. Olugbara, T. B. Taiwo et al., "Alignment-free z-curve genomic cepstral coefficients and machine learning for classification of viruses," in *Proceedings of the International Conference on Bioinformatics and Biomedical Engineering, Granada, Spain*, pp. 290–301, Springer, April 2018.
- [18] T. Shen, Y. Nagai, M. Udayakumar et al., "Automated genomic signal processing for diseased gene identification," *Journal of Medical Imaging and Health Informatics*, vol. 9, no. 6, pp. 1254–1261, 2019.
- [19] G. S. Randhawa, K. A. Hill, and L. Kari, "Ml-dsp: machine learning with digital signal processing for ultrafast, accurate, and scalable genome classification at all taxonomic levels," *BMC Genomics*, vol. 20, p. 267, 2019.
- [20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [21] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: a comprehensive review," *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.
- [22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1097–1105, Lake Tahoe, NV, USA, December 2012.
- [23] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 221–231, 2012.
- [24] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, Columbus, OH, USA, June 2014.
- [25] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5325–5334, Boston, MA, USA, June 2015.
- [26] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2414–2423, Las Vegas, NV, USA, June 2016.
- [27] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV)*, pp. 565–571, IEEE, Stanford, CA, USA, October 2016.
- [28] G. Litjens, T. Kooi, B. E. Bejnordi et al., "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.
- [29] S. Wang, J. Xiang, Y. Zhong, and Y. Zhou, "Convolutional neural network-based hidden markov models for rolling element bearing fault identification," *Knowledge-Based Systems*, vol. 144, pp. 65–76, 2018.
- [30] S. Wang and J. Xiang, "A minimum entropy deconvolution-enhanced convolutional neural networks for fault diagnosis of axial piston pumps," *Soft Computing*, vol. 24, no. 4, pp. 2983–2997, 2020.
- [31] S. Wang, J. Xiang, Y. Zhong, and H. Tang, "A data indicator-based deep belief networks to detect multiple faults in axial piston pumps," *Mechanical Systems and Signal Processing*, vol. 112, pp. 154–170, 2018.
- [32] R. F. Voss, "Evolution of long-range fractal correlations and 1/f noise in DNA base sequences," *Physical Review Letters*, vol. 68, no. 25, pp. 3805–3808, 1992.
- [33] P. Welch, "The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Transactions on Audio and Electroacoustics*, vol. 15, no. 2, pp. 70–73, 1967.
- [34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, Las Vegas, NV, USA, June 2016.
- [35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: a large-scale hierarchical image database," in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, IEEE, Miami, FL, USA, June 2009.
- [36] M. K. Iyer, Y. S. Niknafs, R. Malik et al., "The landscape of long noncoding RNAs in the human transcriptome," *Nature Genetics*, vol. 47, no. 3, pp. 199–208, 2015.
- [37] B. T. Wilhelm, S. Marguerat, S. Watt et al., "Dynamic repertoire of a eukaryotic transcriptome surveyed at single-nucleotide resolution," *Nature*, vol. 453, no. 7199, pp. 1239–1243, 2008.
- [38] A. Sanchez Calle, Y. Kawamura, Y. Yamamoto, F. Takeshita, and T. Ochiya, "Emerging roles of long non-coding RNA in cancer," *Cancer Science*, vol. 109, no. 7, pp. 2093–2100, 2018.
- [39] R. Tripathi, S. Patel, V. Kumari, P. Chakraborty, and P. K. Varadwaj, "DeepLNC, a long non-coding RNA prediction tool using deep neural network," *Network Modeling*



- Analysis in Health Informatics and Bioinformatics*, vol. 5, no. 1, p. 21, 2016.
- [40] S. Han, Y. Liang, Y. Li, and W. Du, "Long noncoding RNA identification: comparing machine learning based tools for long noncoding transcripts discrimination," *BioMed Research International*, vol. 2016, Article ID 8496165, 14 pages, 2016.
  - [41] C. Yang, L. Yang, M. Zhou et al., "LncADeep: anab initio lncRNA identification and functional annotation tool based on deep learning," *Bioinformatics*, vol. 34, no. 22, pp. 3825–3834, 2018.
  - [42] X.-N. Fan and S.-W. Zhang, "lncRNA-MFDL: identification of human long non-coding RNAs by fusing multiple features and using deep learning," *Molecular BioSystems*, vol. 11, no. 3, pp. 892–897, 2015.
  - [43] J. Baek, B. Lee, S. Kwon, and S. Yoon, "LncRNA-net: long non-coding RNA identification using deep learning," *Bioinformatics*, vol. 34, no. 22, pp. 3889–3897, 2018.
  - [44] Z. Cao, X. Pan, Y. Yang, Y. Huang, and H.-B. Shen, "The lncLocator: a subcellular localization predictor for long non-coding RNAs based on a stacked ensemble classifier," *Bioinformatics*, vol. 34, no. 13, pp. 2185–2194, 2018.