WestVirginiaUniversity
THE RESEARCH REPOSITORY @ WVU

Graduate Theses, Dissertations, and Problem Reports

2021

A Deep Learning Approach to LncRNA Subcellular Localization Using Inexact q-mer

Weijun Yi
wy0003@mix.wvu.edu

Recommended Citation

Yi, Weijun, "A Deep Learning Approach to LncRNA Subcellular Localization Using Inexact q-mer" (2021). *Graduate Theses, Dissertations, and Problem Reports*. 10222.
https://researchrepository.wvu.edu/etd/10222

2021

A Deep Learning Approach to LncRNA Subcellular Localization Using Inexact q-mer

Weijun Yi

A Deep Learning Approach to LncRNA Subcellular Localization Using Inexact q-mer

Weijun Yi

A thesis submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University

in partial fulfillment of the requirements for the degree of
Master of Science in
Computer Science

Donald Adjeroh, Ph.D., Chair
Gianfranco Doretto, Ph.D.
Ivan Martinez, Ph.D.

Lane Department of Computer Science and Electrical Engineering
Morgantown, West Virginia
2021

Keywords: deep learning; lncRNA; subcellular localization; q-mer

ABSTRACT

A Deep Learning Approach to LncRNA Subcellular Localization Using Inexact q-mer

Weijun Yi

Long non coding Ribonucleic Acids (lncRNAs) can be localized to different cellular components, such as the nucleus, exosome, cytoplasm, ribosome, etc. Their biological functions can be influenced by the region of the cell they are located. Many of these lncRNAs are associated with different challenging diseases. Thus, it is crucial to study their subcellular localization. However, compared to the vast number of lncRNAs, only relatively few have annotations in terms of their subcellular localization. Conventional computational methods use q-mer profiles from lncRNA sequences and then train machine learning models, such as support vector machines and logistic regression with the profiles. These methods focus on the exact q-mer. Given possible sequence mutations and other uncertainties in genomic sequences and their role in biological function, a consideration of these changes might improve our ability to model lncRNAs and their localization. I hypothesize that considering these changes may improve the ability to predict subcellular localization of lncRNAs. To test this hypothesis, I propose a deep learning model with inexact q-mers for the localization of lncRNAs in the cell. The proposed method can obtain a high overall accuracy of 94.7%, an average of 91.3% on a benchmark dataset, using the 8-mers with mismatches. In comparison, the exact 8-mer result was 89.8%. The proposed approach outperformed existing state-of-art lncRNA predictors on two different datasets. Therefore, the results support the hypothesis that deep learning models using inexact q-mers can improve the performance of computational lncRNA localization algorithms. The lengths of the lncRNAs vary from hundreds to thousands of nucleotides. In this work, I also check whether the length of lncRNA will impact the prediction accuracy. The results show that when the lncRNA sequence's length is between 2000 and 3000 nucleotides, our model is more accurate.

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank my wife. Without her, my academic journey would seem impossible. I also want to thank my parents for their generous support for me.

I also would like to express my deepest appreciation to my advisor, Dr. Donald Adjeroh, for his guidance and suggestion. Without him, I would not finish this work. Meanwhile, I would like to thank my committee members, Dr. Gianfranco Doretto and Dr. Ivan Martinez. Thanks for their time, support, and professional guidance.

The work reported in this thesis was supported in part by grants from the National Science Foundation (Award #s 174778, 1920920).

Table of Contents

List of Table

List of Figure

Chapter 1: Introduction

Non-coding RNAs (ncRNAs) and protein-coding genes are two constituent parts of the human genome [1]. Usually, non-coding RNAs can be divided into small ncRNAs with lengths less than 200 nucleotides and long non-coding RNAs (lncRNAs) with lengths greater than or equal to 200 nucleotides [2]. Since lncRNAs were first discovered in the early 1990s, the family of lncRNAs has expanded rapidly. A recent study indicates that there are over 270,000 lncRNA transcripts in humans [3]. Unlike the protein-coding genes, which are functional units of heredity [4], non-coding RNAs were once deemed non-functional. They were perceived as the product of spurious transcription [5]. However, the application of high-throughput sequencing technologies [6] has shed more light on the transcriptional units. Accumulative evidence shows that ncRNAs, specifically lncRNAs, exhibit biological functions. LncRNAs have been associated with biological processing, such as chromatin modification, cell cycling, protein transcription, and translation [7], [8]. LncRNAs also play essential roles in diseases, including cancer, autism, Alzheimer's disease, and others [9]–[11]. A popular database of lncRNA-associated diseases, LncRNADisease [12], documents 10,564 experimentally supported lncRNA-disease associations. There are 451 unique disease names in the database, including various cancers, syndromes, nervous system disorders, etc., which underline the critical role of lncRNAs in many complex human diseases.

Similar to proteins, the function of lncRNAs has been linked with their subcellular localization in the cell [13]. Therefore, understanding the subcellular localization of lncRNAs and their dynamic changes can also help to explain the function of newly discovered lncRNAs [14]. To study the RNA subcellular localization, a database, RNALocate v2.0 [15], was constructed in 2016 and updated in 2021. 213,260 RNA subcellular localization entries validated by experimental evidence. Experimental results show RNAs can be located in the nucleus, cytoplasm, ribosome, exosome, nucleoplasm, chromatin, cytosol, endoplasmic reticulum, and plasma membrane. See [15]. The dataset contains 9,587 lncRNAs, some of them located in different components of the cell. Only 6728 unique lncRNAs were annotated. In 2017, another database, LncATLAS [16], for subcellular localization of lncRNAs was introduced by calculating the cytoplasmic/nuclear relative concentration index. 6768 lncRNAs were annotated. Compared to the large number of lncRNAs, only a few lncRNAs have been annotated.

Briefly, there are two general approaches to determine the subcellular localization of lncRNA: experimental biomedical methods and computational methods. Several practical methods for lncRNA localization include fluorescence in situ hybridization (FISH), APEX-RIP, Fluorescent In Situ RNA Sequencing, Multiplexed Error-Robust Fluorescence Situ Hybridization, RNA Zipcodes specifying subcellular destinations, etc. [14]. Biomedical experiments determine the subcellular localization by immunolabeling or tagging with a fluorescence microscope. These experiments are time-consuming and laborious [17]. Recent studies indicate that computational approaches can predict subcellular localization by using known subcellular localization datasets. These studies make predictions with high accuracy by extracting shot nucleotide segments (called *q*-mers or *q*-grams) from lncRNA sequences and training machine learning models, such as Random Forest (RF), support vector machines (SVM), or deep neural network models [18]–[20].

Traditional computational methods have focused on exact q-mers. However, given possible mutations in genomic sequences [21] and other uncertainties in biological systems, exact pattern matching may not be adequate to model problems in RNA localization. Thus, segments with inexact matches or mismatch(es) may provide equally biological information in the modeling. In this work, I am interested in whether inexact q-mers can impact the computational prediction of lncRNA localization based on lncRNA sequences. I introduce a deep learning approach for lncRNA localization using inexact q-mers with a one-dimensional convolutional neural network (1D CNN). The length of the lncRNA sequences vary from hundreds to thousands. In this work, I also test whether the length of the lncRNA sequences will impact the prediction on the localization.

Chapter 2: Background and Literature Review

2.1 DNA, RNA, and Gene

Deoxyribonucleic acid (DNA) is the genetic material present in humans and almost all other living things. DNA is a double-helix molecule (nucleotides). It is made up of stacked pairs of nitrogenous bases (adenine (A) and cytosine (C) for purines; guanine (G), or thymine (T) for pyrimidines)[22]. The double-stranded DNA molecule may store genetic information in either strand. A gene is a segment in one of the DNA molecule chains containing genetic data [23], [24]. According to the central dogma of molecular biology, DNA is transcribed into Ribonucleic acid (RNA). Genes served as templates in the synthesis of RNA molecules, the message RNA (mRNA). Message RNA carries the instructions for making proteins, the functional units of the cell, and is eventually converted into protein (translation) by RNA polymerases. Thus, the genetic information flows from DNA to RNA and then to protein. RNA polymerases synthesize RNA molecules by complementing one strand of the DNA with the replacement of thymine (T) by uracil (U)[22].

For decades, RNA was only deemed to be the messenger between DNA and proteins. However, the rapid progress in DNA/RNA sequencing technologies has revealed that most regulatory RNAs' functions do not involve protein translation. Instead, RNAs play roles in the regulation of gene expression, cell cycle control[25], [26].

2.2 Coding vs. Noncoding Genes

After the genes are transcribed into single RNA molecules, mRNA, they are translated into protein. These kinds of genes are protein-coding genes. The intermediate, message RNA, is the protein-coding RNA. There are also genes for RNA molecules that are not translated. These genes are non-coding genes. The RNA molecules produced from non-coding genes are non-coding RNAs (ncRNAs). A genome is the genetic material of an organism. It includes the genes and ncRNAs. For many years, non-coding genes and non-coding RNAs were treated as “genetic junk” and non-functional. In contrast, scientists have proved that non-coding genes and RNAs play vital roles in protein synthesis. Non-coding RNAs help form a protein by assembling the protein building blocks into a chain. In addition, non-coding genes contain many types of regulatory functions, such as assisting activate transcription[5], [27]–[29].

People divide Non-coding RNAs into two major categories: structural or housekeeping RNAs and regulatory RNAs. The transfer RNAs and ribosomal RNAs are structural. They play roles in mRNA translation. The regulatory RNAs are involved in various aspects of cellular processes, from transcriptional regulation to control of translation. Regulatory RNAs usually come in different sizes: small ncRNAs with nucleotides less than 200 and long ncRNAs (lncRNAs) with the length of nucleotides greater or equal to 200. Small ncRNAs include small interfering RNAs (siRNAs) that function as gene regulation, microRNAs (miRNAs) that function as post-transcriptional regulation, piwi-interacting RNAs (piRNAs) that regulate genetic elements in germ cell lines. The rest of the ncRNAs with lengths greater than or equal to 200 are lncRNAs [2], [25], [28], [30].

2.3 Long non-coding RNAs

A conventional opinion is that less than 2% of 3 billion DNA bases of the human genome encode proteins, and most of the other genomes are functionally unknown[31], [32]. Long non-coding RNA is a relatively new class found functional in biological processing among these rest genomes.

Long non-coding RNAs (lncRNAs) are defined as ncRNAs with lengths greater than or equal to 200 nucleotides [2]. Since lncRNAs were first discovered in the early 1990s, the family of lncRNAs has expanded rapidly. A recent study indicated that there are over 270,000 lncRNA transcripts in humans [3]. Unlike the protein-coding genes, which are functional units of heredity[4], non-coding RNAs were once deemed non-functional. They were perceived as the product of spurious transcription [5]. However, the application of high-throughput sequencing technologies [6] has shed more light on the transcriptional units. Accumulative evidence shows that ncRNAs, specifically lncRNAs, exhibit biological functions. LncRNAs have been associated with biological processing, such as chromatin modification, cell cycling, protein transcription, and translation[8], [25]. LncRNAs also play essential roles in diseases, including cancer, autism, Alzheimer's disease, and others [9]–[11].

A popular database of lncRNA-associated diseases, LncRNADisease[12], documents 10,564 experimentally supported lncRNA-disease associations. There are 451 unique disease names in the database, including various cancers, syndromes, nervous system disorders, etc., which underline the critical role of lncRNAs in many complex human diseases.

To fully understand the functionality of lncRNA, it is critical to identify and annotate lncRNA from the sea of human genomes. There are two major kinds of methods to identify the function and mechanisms of lncRNA. The experimental methods identify lncRNA-protein interactions[33]. The experimental methods are time-consuming and professional knowledge is highly demanded. The others are computational methods. These methods extract sequence features relationship from experimentally verified lncRNA-protein interacting pairs and then use machine learning methods or deep learning methods to predict novel lncRNAs[34]–[38].

Similar to proteins, the function of lncRNAs has been linked with their subcellular localization in the cell [13]. Therefore, understanding the subcellular localization of lncRNAs and their dynamic changes can also help to explain the function of newly discovered lncRNAs[39].

2.4 Subcellular localization

RNAs play crucial roles in cellular processes, including translating genetic information, regulating gene activity, and cellular differentiation [40]. These functions are determined by RNAs' location in the cell [14], [41]. The cell of eukaryotic organisms can be divided into functionally distinct membrane-bound compartments [40](See Figure 1.), which are linked with different phases of biological processes[42]. To understand the function of RNA, we need to understand its subcellular localization.

Experiment methods, such as FISH, APEX-RIP, Fluorescent In Situ RNA Sequencing, Multiplexed Error-Robust Fluorescence In Situ Hybridization, RNA Zipcodes Specifying Subcellular Destinations [14], which map RNAs to their subcellular localization, require knowledge of molecular chemistry, specialized instruments, and techniques. These experiments are time-consuming and laborious[17].

Unlike the coding RNAs, which have been studied widely, lncRNAs are more challenging to explore, given their low expression levels [43]. Thus, using information from known datasets to predict the subcellular localization of lncRNAs has become a significant challenge. To study the RNA subcellular localization, a database, RNALocate V2.0 [15], was constructed in 2016 and updated in 2021. It documents 213260 curated RNA subcellular localization entries with experimental evidence. Experimental results show RNAs can locate in the nucleus, cytoplasm, ribosome, exosome, nucleoplasm, chromatin, cytosol, and endoplasmic reticulum.

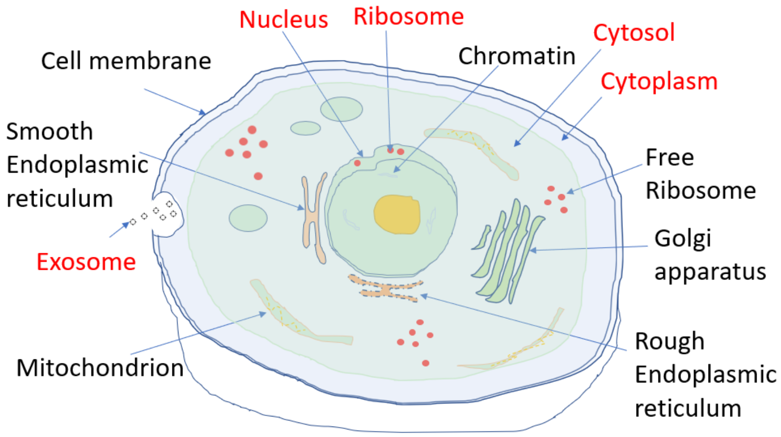


Figure 1. The structure of an animal cell. The key target lncRNA localizations in most datasets are the nucleus, exosome, ribosome, and cytoplasm (indicated in red).

In 2017, another database, LncATLAS [16], was built for the subcellular localization of lncRNAs. 6768 lncRNAs were annotated. Compared to the enormous volume of lncRNAs, the annotated lncRNAs are very small. Most of them are still functionally unknown.

Recent studies indicate that subcellular localization can be predicted from known subcellular localization dataset with computational approaches. Machine learning is applied when making the prediction. These studies annotate lncRNAs with subcellular localizations, such as cytoplasm, nucleus, ribosome, exosome, etc. They extract pseudo nucleotide (q-mer) segments from lncRNA sequence and train Random Forest (RF), support vector machine (SVM), or deep neural network models[18]–[20].

From known datasets to predict the new lncRNAs' subcellular localization is becoming a hot issue. With known annotation, prediction on subcellular localization can be treated as a classification problem. For coding RNAs, there are many predictors of protein, which have been developed since 1990s[39]. Many take computational approaches, using machine learning or deep learning methods. In contrast to protein-coding RNAs, only a few methods have been proposed predicting lncRNAs subcellular localization.

2.5 Prior work on lncRNA subcellular localization

Research shows that we can represent RNA sequence in a discrete model: pseudo-k-tuple nucleotide composition (PseKNC) [44]. In the PseKNC model, *q*-length substrings (q-mers) are extracted from the RNA sequence. Each substring can be treated as an RNA motif that contains some biological information. Then, the RNA sequence is decomposed into a set of small-sized segments, which are typically more efficient to analyze than long RNA sequences. Along this line of thought, Kirk et al. [45] showed that profiles based on such q-mers could be used to analyze lncRNAs subcellular localization. They create q-mer profiles for all lncRNAs in human and mouse GENECODE databases[46]. They compare the similarity between lncRNA sequences by computing the Pearson's correlation of the sequences' q-mer profiles and dividing lncRNAs into five communities. By analyzing the distributions of nuclear ratios between communities, they conclude that q-mer content provides information about the lncRNA subcellular localization. These make it reasonable to annotate subcellular localization of new lncRNA from the known database with a computational approach.

General computational methods predict the localization of lncRNAs by extracting q-mer features from the lncRNA sequence. They select particular nucleotide segments as features and then train a prediction model, such as random forest, support vector machine, or deep neural network, to make a prediction.

In LncLocator[18], Cao *et al.* created an annotated subcellular localization dataset of lncRNAs from RNALocate [47]. The dataset contained 612 lncRNAs which are allocated in 5 locations in the cell, including nucleus, cytoplasm, cytosol, ribosome, and exosome (see Table 1). They extracted q-mer (q=4,5,6) segments from lncRNA sequences. Considering the low discrimination of small segments, they feed 4-mer features into a stacked autoencoder model to automatically create high-level feature representation. Since the dataset is imbalanced, they used a supervised over-sampling strategy to expand the minority sample size. They train machine learning models, namely, random forest and support vector machines with raw *q*-mer features and autoencoder-based features. These resulted in four prediction models, which are then combined using a stacked ensemble model for final prediction. They tested their data in various scenarios. The overall accuracy was 59.8% on the five-class dataset.

The iLoc-lncRNA [19] predicts lncRNAs subcellular localization by feeding octamer features into SVM. They build a 4-class dataset from RNALocate [47]. The classes correspond to the following localizations: nucleus, cytoplasm, ribosome, and exosome. There are 655 lncRNAs in the dataset (see Table 1). First, they extract 8-mer features from the lncRNA sequences. Then, because high dimensional features will produce several problems such as over-fitting and redundant noise, they selected features based on the 8-mer feature distribution probability. They finally picked 4107 8-mer features and then trained the SVM model with the extracted features. The overall accuracy was 86.72% on the 4-class dataset.

Gudenas et al. [20] built a two-class dataset from the ENCODE [31] project. First, they quantified the lncRNA transcript differences between nuclear and cytosolic, applying $\log_2$ fold-change threshold to allocate 8678 lncRNAs to cytosolic and nuclear, 4380 for cytosolic, and 4298 for nuclear. They then extracted q-mer features (q=2,3,4,5) from the lncRNA sequences. Next, they added RNA-protein binding motifs to the feature map, and passed these to a deep neural network. They obtained an accuracy of 72.4%.

In lncLocPred [48], built a four-class dataset from the RNALocate database [47]. The database contains 396 lncRNAs. They use this dataset as an independent dataset and dataset in iLoc-lncRNA

[19] as the benchmark dataset. First, they collect features using *q*-mers (*q*=5,6,8), triplet, and PseDNC. They then trained a logistic regression model using the selected features. They finally got highest accuracy of 92.37%.

In Locator-R [49], Ahmad et al. use the n-gapped *l*-mer composition and *l*-mer composition as features and train support vector machines. As a result, they get an overall accuracy of 90.09%.

In KD-KLNMF, they introduce a novel statistical model using k-mer incorporated with dinucleotide-based spatial autocorrelation as the feature map and apply synthetic Minority over-sampling technique to deal with the imbalance dataset. They then train support vector machines and get an overall accuracy 97.24%.

These methods both use the data from the iLoc paper as a benchmark dataset; the KD-KLNMF is slightly different. They both focus on the exact q-mer profiles. However, they apply various feature selection methods, and both get excellent performance.

	Recent computation-based approaches to LncRNA Localization						
	LncLocator [18]	iLoc-lncRNA[19]	lncLocPred[48]	Locate-R[49]	KD-KLNMF[50]	Gudenas[20]	
Nucleus	152	156	156	156	154	cytosolic	4380
Cytoplasm	301	426	426	426	417	nuclear	4298
Cytosol	91	---	---	---	---		
Ribosome	43	43	43	43	43		
Exosome	25	30	30	30	30		
Total	612	655	655	655	644		8678
OA(%)	66.5	86.72	92.37	90.69	97.24		72.4

Table 1.Recent computation-based approaches to LncRNA Localization

Chapter 3 Methodology

This work examines the impact of inexact q-mer profiles on the prediction performance on multi-label lncRNA subcellular localization. In this paper, both exact and inexact q-mer profiles are extracted from the lncRNA sequences to build feature maps, and then a 1D convolutional neural network (1D CNN) model is trained. To compare the performance of this method with the existing state-of-the-art techniques, we use the datasets from LncLocator [18]with 5-components and iLoc-lncRNA [19]. The workflow is as Figure 2 shows. First, I will extract the feature map from the lncRNA sequences, then apply data preprocessing, feature selection, and finally, feed our preprocessed data into a 1-dimensional convolutional neural network to predict localization.

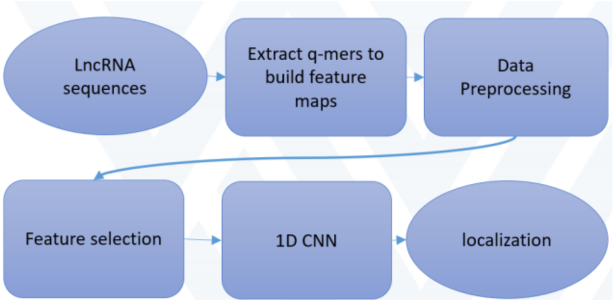


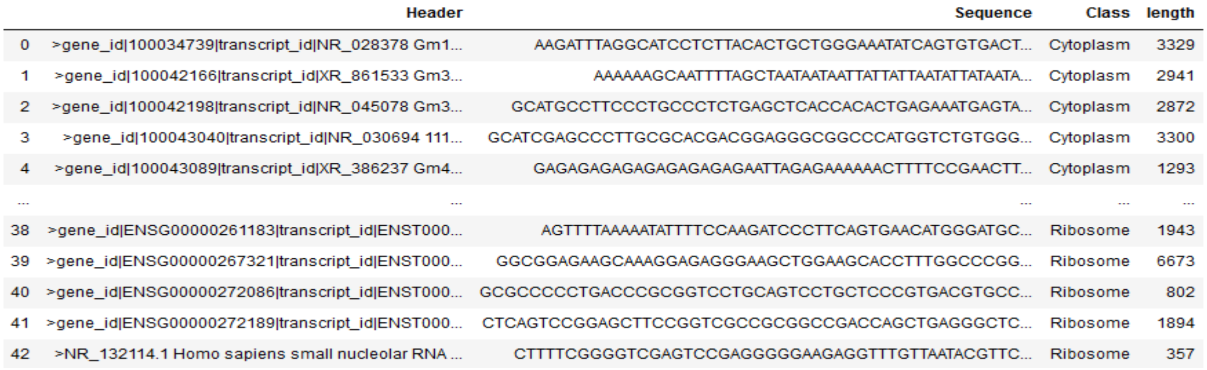
Figure 2. Wrokflow of this work.

3.1 Localization as a classification problem

The datasets of lncRNAs subcellular localization have been annotated by experimental methods. Each lncRNA sequence is linked to one location in the cell. We can treat the localization as a supervised classification problem.

3.2 Dataset

I tested the datasets from the lncLocator [18], iLoc-lncRNA [19]. Both of them obtained lncRNA sequences from RNALocate [47]. Four subcellular localizations (classes), nucleus, cytoplasm, exosome, and ribosome, are retained in iLoc-lncRNA. And five in lncLocator, nucleus, cytoplasm, cytosol, exosome, and the ribosome. The subcellular localizations include the nucleus, cytoplasm, cytosol, ribosome, and exosome (See Figure 1). The dataset is as Table 2 shows. Each row is a lncRNA sequence. The first column is the information of lncRNA. The second column is the sequence of lncRNA. The third column is the location of lncRNA in the cell. The Fourth column is the length of the lncRNA. The lengths of the lncRNA sequences vary from hundreds to thousands of nucleotides. I divided the lncRNA sequences into four groups. The length is less than 1000 nucleotides, between 1000 and 2000 nucleotides, between 2000 and 3000 nucleotides, and greater than 3000 nucleotides. The lncRNA sequence distribution is as table 3 shows.



	Header	Sequence	Class	length
0	>gene_id|100034739|transcript_id|NR_028378 Gm1...	AAGATTTAGGCATCCTCTTACACTGCTGGGAAATATCAGTGTGACT...	Cytoplasm	3329
1	>gene_id|100042166|transcript_id|XR_861533 Gm3...	AAAAAAGCAATTTTAGCTAATAATAATTATTATTAATATTATAATA...	Cytoplasm	2941
2	>gene_id|100042198|transcript_id|NR_045078 Gm3...	GCATGCCTTCCCTGCCCTCTGAGCTCACCACACTGAGAAATGAGTA...	Cytoplasm	2872
3	>gene_id|100043040|transcript_id|NR_030694 111...	GCATCGAGCCCTTGCGCACGACGGAGGGCGGCCCATGGTCTGTGGG...	Cytoplasm	3300
4	>gene_id|100043089|transcript_id|XR_386237 Gm4...	GAGAGAGAGAGAGAGAGAGAATTAGAGAAAAAACTTTTCCGAACTT...	Cytoplasm	1293
...	...	...	...	...
38	>gene_id|ENSG00000261183|transcript_id|ENST000...	AGTTTTAAAAATATTTTCCAAGATCCCTTCAGTGAACATGGGATGC...	Ribosome	1943
39	>gene_id|ENSG00000267321|transcript_id|ENST000...	GGCGGAGAAGCAAAGGAGAGGGAAGCTGGAAGCACCTTTGGCCCGG...	Ribosome	6673
40	>gene_id|ENSG00000272086|transcript_id|ENST000...	GCGCCCCCTGACCCGCGGTCCTGCAGTCCTGCTCCCGTGACGTGCC...	Ribosome	802
41	>gene_id|ENSG00000272189|transcript_id|ENST000...	CTCAGTCCGGAGCTTCCGGTCGCCGCGGCCGACCAGCTGAGGGCTC...	Ribosome	1894
42	>NR_132114.1 Homo sapiens small nucleolar RNA ...	CTTTTCGGGGTCGAGTCCGAGGGGGAAGAGGTTTGTTAATACGTTC...	Ribosome	357

Table 2. dataset of the lncRNA subcellular localization. See text.

Length of lncRNA	Number of sequences	
	lncLocator[18]	iLoc[19]
length < 1000	142	154
1000 ≤ length < 2000	185	215
2000 ≤ length <3000	142	146
length > 3000	143	140
Total	612	655

Table 3. The lncRNA sequence length distribution of the datasets.

3.3 Feature representation

LncRNA is transcribed from DNA. LncRNA consists of *a* string of nucleotides bases. These bases are adenine (A), guanine (G), uracil (U), and cytosine (C). The sequence of lncRNA can be represented as:

$$S = S_1S_2\ldots S_i\ldots S_n, \text{ with } S_i \in \{A, G, C, U\}$$

Here n is the length of the sequence, and S_i is i^{th} nucleotide base, $1 \leq i \leq n$.

3.3.1 q-mer profile

The *q*-mer is a substring of a sequence with length *q*. A possible q-mer will be a q-length substring with one of the A, C, G, U symbols from a lncRNA sequence. There are 4^q possible different *q*-mers in one lncRNA sequence. We build the feature map with the *q*-mer profile, which captures the probability distribution (or frequency of occurrence) of each given possible *q*-mer. For the *q*-mer profile, typically, each row corresponds to one lncRNA, and each column corresponds to one of the possible *q*-mers. Each cell is a feature value that represents the frequency of the *q*-mer in the given sequence. The class is the localization of the lncRNA sequence. We compute the feature value by running a *q*-length window with stride one across the sequence. If the segment is in the sequence, then the frequency of the segment is set to its feature value. Otherwise, the feature value is 0. Based on this, we define the feature map (FM) of a lncRNA sequence as FM(S) = $\{Q_i: f_i\}$, $1 \leqslant i \leqslant N\}$, Here *S* is the sequence, Q_i is the i^{th} *q*-mer, f_i is the corresponding feature value, and *N* is the number of possible unique *q*-mers in the sequence. For example, we can compute the 3-mer feature map for the sequence **S=*AGCUAGUA***. First, we find all the 3-mer combinations of A, G, C, and U. Then, we map the frequency of each 3-mer. Finally, we get the feature map: FM(**S**)={AAA:0, AAG:0, ▪▪▪, AGC:1, ▪▪▪, AGU:1, ▪▪▪, UUU:0}. The feature map of the dataset will be as Table 4 shows.

	AAA	AAG	•••	AGU	•••	UUU	Class
lncRNA 1	0	0	•••	1	•••	0	Nuclear
lncRNA 2	2	4	•••	5	•••	3	Ctytoplasm
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
lncRNA n	2	5	•••	7	•••	9	Exosome

Table 4. Feature map of the dataset (Using 3-mer, for example).

3.3.2 Inexact q-mer profiles.

In this work, I introduce the idea of inexact *q*-mer profile. I focus on the *q*-mers with *k*-mismatch(es), also called the (*q*, *k*)-mismatch kernel, which provides the idea of mismatching in biological interest [51], [52]. Given a *q*-mer, we compute the frequency of other matching *q*-mers, where a match is allowed to admit at most *k*-mismatches, here $k < q$. Each matching *q*-mer is still required to have the same length of *q*, just like the given *q*-mer. Thus, for a given *q*-mer, say *Q*, the result of (*q*, *k*)-mismatch is thus a set of q-mers, such that each feature (*q*-mer) in the collection has the same length as *Q*, and there are at least *q*-*k* base(s) that have an exact match with bases in the given *q*-mer, *Q*. For example, for the *q*-mer sequence **Q=*AGCUAGUA***, *the* (8, 1)-mismatches are shown in Table 5. We use the hamming distance to measure the mismatch. Row 1 is the original sequence. From rows 2 to 9, each row denotes a mismatch that happened at a different location of the original sequence. The asterisk indicates one of three bases that is other than the original base, respectively. Thus, for each row, there are three possible mismatch *q*-mers. Hence, in this example, there are 24 possible mismatch *q*-mers. We then set the frequency value of 8-mer, ***AGCUAGUA,*** with 25 (24 mismatches + 1 match). This work uses a naïve method to compute the (*q*, *k*)-mismatch feature map. There exist efficient data structures using suffix trees and suffix arrays[53] to calculate the feature map.

1	A	G	C	U	A	G	U	A
2	*	G	C	U	A	G	U	A
3	A	*	C	U	A	G	U	A
4	A	G	*	U	A	G	U	A
5	A	G	C	*	A	G	U	A
6	A	G	C	U	*	G	U	A
7	A	G	C	U	A	*	U	A
8	A	G	C	U	A	G	*	A
9	A	G	C	U	A	G	U	*

Table 5. (8, 1) mismatch for example

A naïve method to compute the (q, k) mismatch feature map is as follows:

```
Feature counts = M x N matrix (M is the number of lncRNAs, N is the number of
possible q-mer profiles, N = $4^q$.  Column name is the q-mer profile.)
Increment = 1
For sequence in dataset:
    counts = default dictionary
    Length = sequence.length
    For i in range (1, Length-q+1):
        qmer = sequence[i:i+q]
        if qmer in counts:
            Continue
        else:
            for j in range(1, length-q+1):
                if hamming_distance (qmer, sequence[j:j+q])  <= k:
                    counts[qmer] += Increment
Map counts to Feature counts
```

3.3.3 Data preprocessing

The feature maps of the lncRNA sequences in the two datasets are counts of the q-mers. First, I normalized the counts according to the length of lncRNA sequences, respectively. I then split the dataset into training and testing sets with a ratio of 4: 1 and did z-score normalization on the training and test sets. The formula is as follows: $z_i = (x_i - \mu)/\sigma$. Here z_i is the score of i-th q-mer feature, x_i is the count of q-mer, μ is the mean q-mer count of all lncRNA sequences, and σ is the standard deviation.

3.3.4 Feature selection.

The dimension of the feature map is 4^q. It grows exponentially when q increases. A high-dimension features map means more noise, which will reduce the accuracy of a predictor. Second, a high dimension will lead to over-fitting [54], which does not accurately predict. Third, high dimensional features will exhaust the computational capacity. I test 3 to 8-mer and some q-mer fusions (see Table 6). The dimension of 7-mer is 4^7, that is 16384, and 8-mer is 65536. For 7-mer, to reduce the feature map size, we applied the X^2 test feature selection method from scikit-learn [55] to get a feature rank and then selected the optimal subset. I started with a subset with the first feature in the rank and added eight features into the subset per time. I tested the performance of the model and took the subgroup with the highest accuracy. I tried the 4107 features in the iLoc-lncRNA [19] dataset on 8-mer.

Feature	Dimension of feature
3-mer	64
4-mer	256
5-mer	1024
6-mer	4096
7-mer	1120/7976*
8-mer	4107**
3 and 4-mer fusion	310
4 and 5-mer fusion	1280
5 and 6-mer fusion	5120
3, 4, and 5-mer fusion	1344
3, 4, 5, and 6-mer fusion	5440

Table 6. Feature Dimension. For 7-mer, I use 1120 features on the lncLocator dataset and 7976 on iLoc dataset. The iLoc paper post 4107 features on 8-mer. I will test these features.

3.3.4 Data preprocessing

The feature maps of the lncRNA sequences in the two datasets are counts of the q-mers. First, I normalized the counts according to the length of lncRNA sequences, respectively. I then split the dataset into training and testing sets and did z-score normalization on the training and test sets. The formula is as follows: $z_i = (x_i - \mu)/\sigma$. Here z_i is the score of i-th q-mer feature, x_i is the count of q-mer, μ is the mean q-mer count of all lncRNA sequences, and σ is the standard deviation.

3.4 Deep learning architecture

3.4.1 1D CNN

The convolutional neural network is a class of deep neural networks that employs a mathematical operation called convolution in at least one of its layers [54]. With convolution, a new feature map from the input feature is detected. Unlike 2D CNN, which is broadly used to operate 2-dimension data such as images and videos, 1D CNN is designed to operate one-dimensional signals such as time series digital signal processing (DSP). It is used in time domain analysis and frequency domain analysis. A typical 1D CNN architecture includes an input layer, convolutional layers (feature extractor section), Multilayer Feed Forward (MLFF) layers (classification section), and output layer [56].

3.4.2 Elements of 1D CNN

A typical 1D CNN architecture includes an input layer, convolutional layers (feature extractor section), Multilayer Feed Forward (MLFF) layers (classification section), and output layer. In addition, there are several operations and terminologies involved in the CNN structure.

The essential element of the network is convolution. In mathematics, applying convolution to one function will change its shape. For a given lncRNA sequence, we can define 1D convolution as S = f * g, S denotes the convolution, f is the input lncRNA sequence, g is the kernel, and * is the convolution operation. In CNN, the kernel is a feature detector. It is a vector of weights of the model. The output S is also known as the feature map [54]. Suppose the length of the lncRNA sequence is m, and the size of the kernel is n, then the element of S, $S(i) = \sum_{k=1}^{n}(f(i + k - 1)\ X\ g(k))$, here $1 \leq i \leq$ m-n. The figure shows the process computing convolution. f is input lncRNA sequence, from f1 to f8 are the features in f. g is the kernel, the kernel size is 3 x 1. From g1 to g3 are the weights in the kernel. According to the definition, the results of convolution S(1)

$= f_1 \times g_1 + f_2 \times g_2 + f_3 \times g_3$. (See figure 3). We move the window 1 step (the stride) along the sequence to compute S(2) until the window reaches the end of the lncRNA sequence. The shape of S is m – n + 1 = 8-3+1 = 6. The value in the two ends, f1 and f8, is used only once. When doing convolution, we may lose some information on the border. To keep the information, we add 0's to both ends. We call this method padding. With padding, the output shape of convolution is the same as the input.

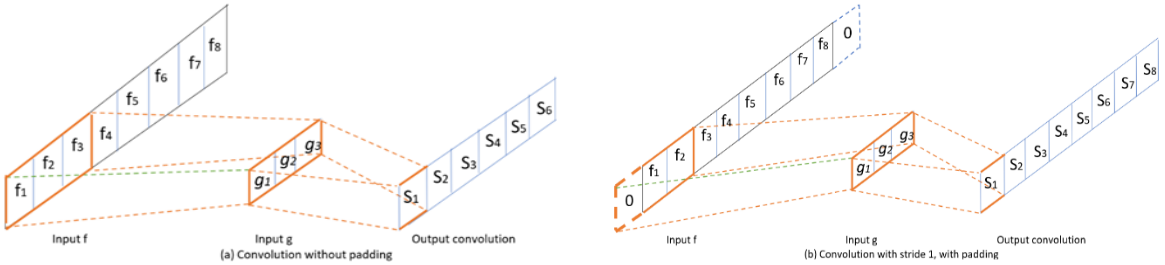


Figure 3. (a) shows the convolution without padding. (b) shows the convolution with padding.

After convolution, the CNN model uses an activation function, called Rectified Linear Unit (ReLU) [54], to the output of convolution. The function is defined as f(x) = max(0, x).

To prevent overfitting [54], we usually apply a pooling layer and dropout operation after the convolution layer. In this paper, we use max-pooling. Max-pooling reduces the feature map size by taking the maximum value of the elements in a stride window. For example, the max-pooling result of digital sequence 2,3,5,1,2,6,8,5, with stride 2, is 3,5,6,8. Dropout is used to randomly set the weights of some neural units in the network to 0 during training time. We also take early stopping [57] to prevent overfitting.

After several convolution layers, the output feature map is a multiple dimension matrix. Then, we need to transform them into a vector with the required output shape. First, we apply a fully connected layer (flatten), which converts the output matrix of convolution lays into a 1-

dimensional vector. And then, a dense layer is used to change the dimension of the result of flattening.

The output layer applies the softmax activation function to map the output probability of each class in the range [0, 1]. The sum of the possibilities of all the classes is 1. For a multiclass classification task, output n-dimension vector with one-hot encoding. One-hot encoding represents categorical variables with binary vectors. The categorical values are mapped to integer values. In binary vector, all the values are zeros except the according integer is 1. For 4-class, one-hot encoding can be represented as:

Class 1: [1,0,0,0]
Class 2: [0,1,0,0]
Class 3: [0,0,1,0]
Class 4: [0,0,0,1]

3.4.3 Architecture used in paper

The feature map of the lncRNA dataset has two attributes: 1) The feature is with a fixed length, and 2) only the feature frequency is considered. Furthermore, the location of each q-mer feature is ignored. Thus, a 1D CNN model is suitable for this scenario. In this work, I build the 1D CNN model with Keras [58].

Figure 4 shows the proposed architecture for the 1D CNN model. Using the 6-mer profile, I can describe the CNN model architecture for a 4-classes task as follows:

1) **Input layer**: hold the raw values of q-mer features with the input shape 4096 x 1.
2) **1st Convolution layer:** 64 filters, kernel-size is 3x1, padding='same', stride=1, followed by Rectified Linear Unit (ReLU) operation. The output feature maps are 64×4096×1.
3) The 1st convolution layer is followed by Max-Pooling operation, with pool size 2 and 25% dropout to avoid overfitting. The output feature maps 64×2048×1.

4) **2nd Convolution layer:** 128 filters, kernel-size is 3x1, padding='same', stride=1, followed by ReLU operation. The output feature map is 128×2048×1.

5) The 2nd convolution layer is followed by Max-Pooling operation, with pool size 2 and 25% dropout. The output feature maps 128×1024×1.

6) **3rd Convolution layer:** 256 filters, kernel-size is 3x1, padding='same', stride=1, followed by ReLU operation. The feature map is 256×1024×1.

7) The 3rd convolution layer is followed by Max-Pooling operation, with pool size 2 and 25% dropout. The output feature maps are 256×512×1.

8) After the convolution layer, there is a flatten layer. The flattening layer transforms the entire pooled feature map matrix into a single column. We then will feed this column to the neural network for processing. The output shape of the feature map is 131072 x 1.

9) We then use a dense layer, followed by ReLU, which serves as a classification section. The dense layer is followed by a 25% dropout. The output shape of the feature map is 32 x 1.

10) **Output layer.** A nonlinear Softmax operation was applied. Finally, output the results of classification.

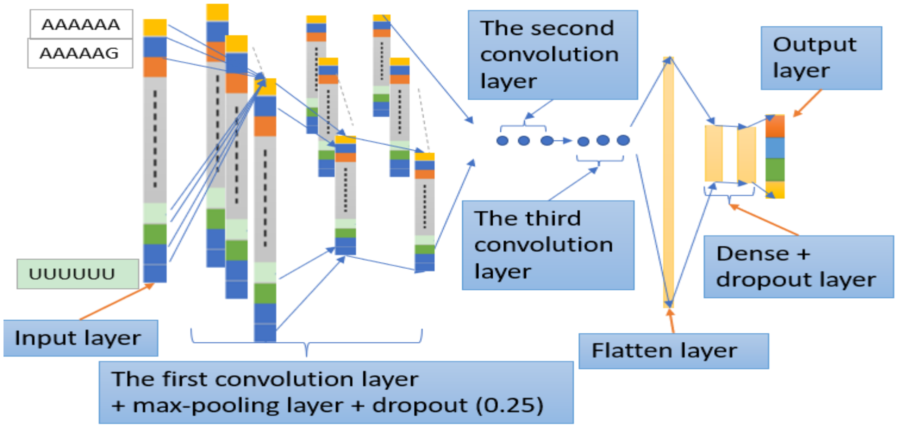

AAAAAA
AAAAAG
UUUUUU
Input layer
The first convolution layer + max-pooling layer + dropout (0.25)
The second convolution layer
The third convolution layer
Flatten layer
Dense + dropout layer
Output layer

Figure 4. 1D CNN architecture used in this work. We were using 6-mer profile as an example. The second and third convolution layer blocks have the same structure as the first one: a convolution layer followed by a max-pooling layer with 0.25 dropout. See text.

3.5 Evaluation

I use overall accuracy (OA), sensitivity (Sn), specificity (Sp), Matthew's correlation coefficient (MCC) and F1-score to evaluate the performance of the 1D CNN model, which are computed with the equation OA = $(TP + TN) / (TP + TN + FP + FN)$, Sn = $TP / (TP + FN)$, Sp = $TN / (TN + FP)$, MCC = $(TP \times TN - FP \times FN) / sqrt ((TP+FP)(TP+FN)(TN+FP)(TN+FN))$, F1-score = $2TP/(2TP+FP+FN)$. Here *TP* is the true positive, the number of positive samples we predict correctly. *TN* is the true negative, the number of negative samples we predict correctly predicted. *FP* is false positive, the number of negative samples we incorrectly predict as positive. *FN* is the false negative, the number of positive samples predicted as negative, and sqrt is the square root.

Chapter 4 Experiments & Results

I tested 1D CNN model on the 5-class dataset from the lncLocator [18] and the 4-class dataset from the iLoc [19]. I tried some q-mer combinations and q-mer (q=3, 4, 5, 6, 7, 8) with various mismatches. Given the randomness of the CNN model [59], I did experiments ten times for different scenarios on the two datasets. Finally, I calculated the average of the performance. The results showed that our model performed better on the 4-class iLoc dataset than the 5-class lncLocator dataset, and the q-mer with mismatch(es) could improve the classification performance.

4.1 Results on exact q-mer

The overall accuracy using exact q-mers on the two datasets is shown in Table 6. The table shows that our model performed better on the 4-class (iLoc-lncRNA) dataset, when compared with the 5-class (lncLocator) dataset. The reason is that the lncLocator dataset has 5 class which has 1 more class than the iLoc dataset, while the total number of lncLocator is 612, which is less than iLoc. It means there will be more noise than the iLoc dataset for each class in the lncLocator dataset. Meanwhile, few data can hard to extract sufficient information to make an accurate prediction.

With q increasing, the overall accuracy on the iLoc-lncRNA dataset rose from around 64% (for 3-mer) to 89.85% (for 8-mer), and from about 53% (for 3-mer) to 71.38% (for 8-mer) on the lncLocator dataset. It indicates that the longer the segments might provide more discriminative features for determining the lncRNA subcellular localization. Combining different q-mers using fusion did not seem to improve the result (Table 7).

	3 mer	4 mer	5 mer	6 mer	7 mer	8 mer
iLoc-lncRNA	64.35	64.89	64.27	65.5	68.78	89.85
lncLocator	53.33	54.72	56.1	54.96	53.58	71.38

Table 7. Performance of exact q-mers, showing overall accuracy(%).

Figure 5 shows an upward trend. The proposed method did a better job on four classes dataset than the five classes dataset.

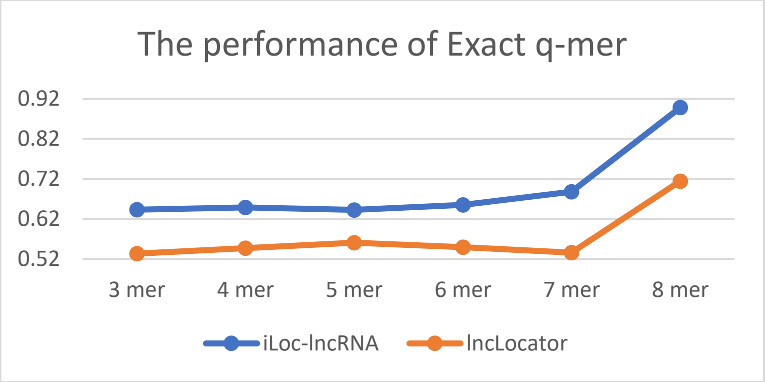


Figure 5. The performance of exact q-mer shows upward trend.

4.2 Results on combined q-mer

The results (Table 8) show that when different lengths q-mer are combined, the model's performance has slight changes.

	34mer	45mer	56mer	345mer	456mer	3456mer
iLoc-lncRNA	64.35	64.89	65.42	65.95	65.11	65.42
lncLocator	55.61	55.53	54.47	55.69	54.88	55.2

Table 8. Performance on combined q-mers

Figure 6 shows there is no significant change when using a different q-mer combination.

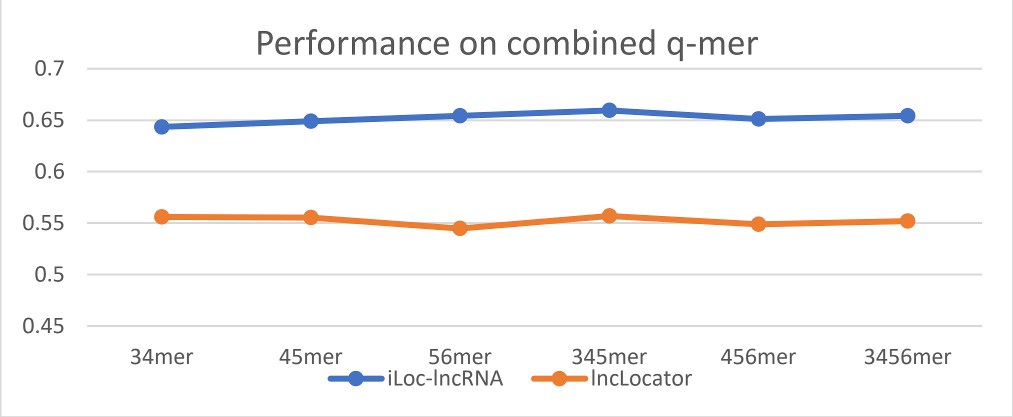


Figure 6. Performance on combined q-mers

4.3 Results on inexact q-mer

4.3.1 Results using q-mers with k-mismatch

I experimented on *q*-mers (*q*=3, 4, 5, 6, 7, 8) with *k* mismatches (0 ≤ k ≤ q-1) on the lncLocator dataset. Table 9 shows the overall accuracy (mean ± standard deviation) when using *q*-mer with *k* mismatch(es) on the lncLocator [18] dataset. The table shows the overall accuracy of the prediction in different scenarios. For example, we can have the utmost seven mismatches in 8-mer. When the number of mismatches is greater than 2, there is a slight improvement in the overall accuracy. The overall accuracy increases from 0.714 to 0.734. However, for smaller *q*-mers, e.g., *q*=3, 4, and 5, increasing the number of mismatches did not necessarily lead to increased accuracy and perhaps add more noise in the model. This may point to the need for a more detailed study of the interplay between *q* and *k* in this (*q*, *k*)-mismatch model.

	0 miss	**1 miss**	**2 miss**	**3 miss**	**4 miss**	**5 miss**	**6 miss**	**7 miss**
8 mer	0.714±0.03	0.703±0.03	0.711±0.045	0.734±0.024	0.72±0.037	0.711±0.035	0.726±0.026	0.716±0.04
7 mer	0.536±0.044	0.521±0.025	0.548±0.036	0.555±0.037	0.554±0.028	0.542±0.033	0.554±0.036	
6 mer	0.55±0.03	0.55±0.03	0.546±0.036	0.534±0.031	0.52±0.049	0.566±0.037		
5 mer	0.561±0.033	0.55±0.047	0.536±0.03	0.524±0.031	0.548±0.031			
4 mer	0.547±0.019	0.518±0.033	0.531±0.028	0.509±0.026				
3 mer	0.533±0.034	0.531±0.02	0.515±0.028					

Table 9. Performance of q-mers with mismatch(es) on the 5-class lncLocator dataset.

Figure 7 shows that 6, 7, and 8-mer have an upward trend.

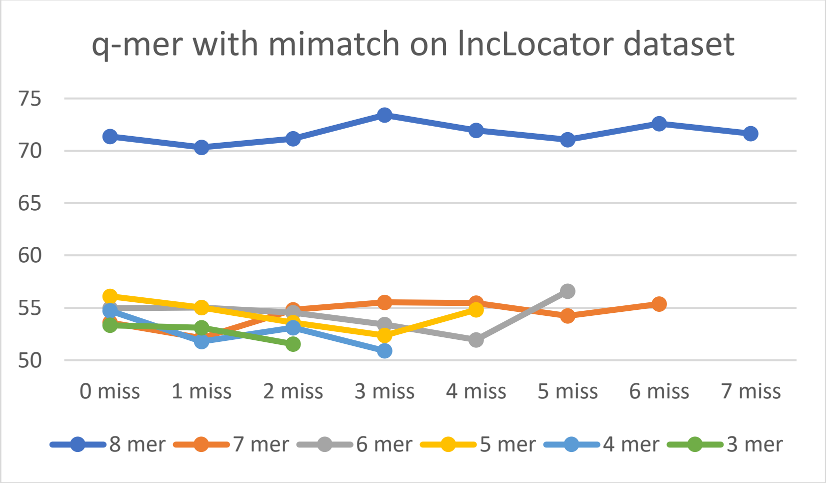


Figure 7. Performance of q-mers on lncLocator dataset.

The proposed method can get the highest overall accuracy of 0.772 (see Table 10) and an average of 0.734 with a standard deviation of 0.024 (see Table 11) for ten times tests on the 5-class lncLocator dataset when taking 8-mer with 3 mismatches.

	Sn	Sp	Preci	MCC	F1-score	OA
Nucleus	0.806	0.913	0.758	0.705	0.781	0.772
Cytoplasm	0.900	0.762	0.783	0.667	0.837	
Ribosome	0.778	0.982	0.778	0.760	0.778	
Exosome	0.400	0.992	0.667	0.501	0.500	
Cytosol	0.389	0.981	0.778	0.502	0.519	

Table 10. The best performance on the lnLocator dataset.

	Sn	Sp	Preci	MCC	F1-score	OA
Nucleus	0.748±0.106	0.928±0.042	0.794±0.083	0.692±0.054	0.762±0.042	0.734±0.024
Cytoplasm	0.885±0.047	0.719±0.087	0.755±0.052	0.615±0.055	0.813±0.024	
Ribosome	0.767±0.161	0.975±0.02	0.736±0.163	0.723±0.13	0.735±0.122	
Exosome	0.34±0.237	0.991±0.012	0.553±0.331	0.403±0.224	0.387±0.221	
Cytosol	0.3±0.221	0.958±0.031	0.428±0.256	0.292±0.214	0.34±0.228	

Table 11. The average performance on the lncLocator dataset.

4.3.2 Results on iLoc-lncRNA dataset

Table 12 shows the corresponding results for the iLoc dataset [19]. We can see the overall accuracy is improved with *k* > 3 using 8-mers. The highest score is 0.921±0.024 when using 8 mer with 6 mismatches, which is 0.028 higher than the exact 8-mer. There are also significant improvements in using *q*= 5, 6, 7 with the *k*-mismatches. Thus, we can conclude that *q*-mer with mismatches performs better than the exact *q*-mer on this dataset.

	0 miss	1 miss	2 miss	3 miss	4 miss	5 miss	6 miss	7 miss
8 mer	0.893±0.025	0.899±0.02	0.891±0.021	0.91±0.017	0.895±0.032	0.901±0.029	0.921±0.024	0.916±0.021
7 mer	0.688±0.031	0.702±0.016	0.711±0.018	0.706±0.028	0.714±0.027	0.71±0.042	0.705±0.03	
6 mer	0.655±0.023	0.659±0.023	0.673±0.035	0.663±0.023	0.653±0.022	0.652±0.034		
5 mer	0.643±0.017	0.651±0.031	0.655±0.013	0.663±0.023	0.649±0.031			
4 mer	0.649±0.011	0.638±0.017	0.644±0.023	0.633±0.037				
3 mer	0.644±0.02	0.644±0.016	0.65±0.004					

Table 12. Performance of q-mers with mismatch(es) on the 4 class iLoc dataset.

Figure 8 visualizes the q-mer with mismatches results on the iLoc dataset. We can see the improvements on 5, 6, 7, 8-mer.

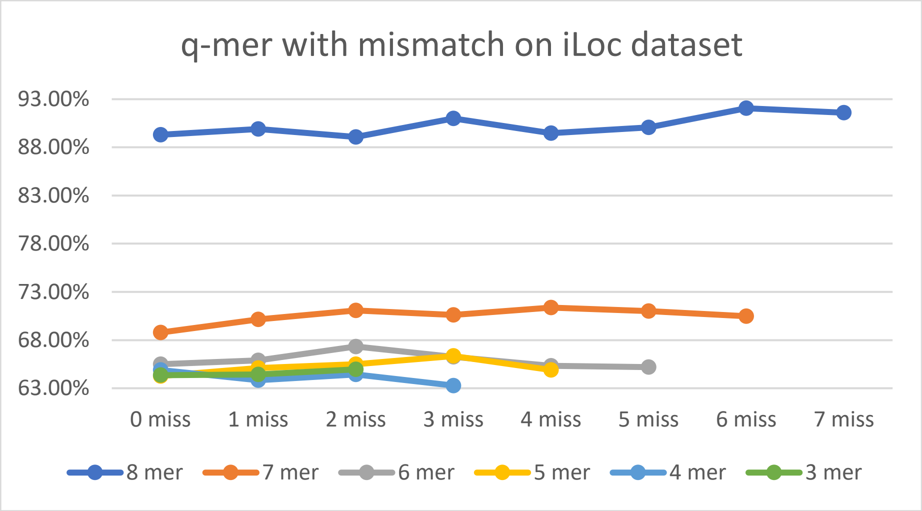


Figure 8. Performance of q-mers with mismatch(es) on iLoc dataset.

The proposed method can get a maximum accuracy of 0.947 (See Table 13) and an average of 0.921±0.024 (See table 14) of 10 times tests on the iLoc-lncRNA dataset when we take 8-mer with 7 mismatches.

	Sn	Sp	Preci	MCC	F1-score	OA
Nucleus	0.935	0.970	0.906	0.896	0.921	94.7
Cytoplasm	0.988	0.913	0.955	0.916	0.971	
Ribosome	1.000	1.000	1.000	1.000	1.000	
Exosome	0.333	1.000	1.000	0.568	0.500	

Table 13. The best performance on te iLoc dataset using 8 mer with mismatches.

	Sn	Sp	Preci	MCC	F1-score	OA
Nuclear	0.884±0.048	0.975±0.012	0.917±0.038	0.87±0.046	0.9±0.036	0.921±0.024
Cytoplasm	0.962±0.02	0.861±0.038	0.928±0.019	0.839±0.042	0.945±0.014	
Ribosome	0.922±0.071	0.992±0.01	0.907±0.108	0.905±0.066	0.909±0.063	
Exosome	0.517±0.229	0.996±0.005	0.79±0.3	0.622±0.243	0.612±0.243	

Table 14. The average performance of 8 mer with 7 mismatches.

4.4 Comparison with existing state-of-the-art predictors

Table 15 exhibits the difference between the proposed method and lncLocator [18]. The proposed method did a good job on the 5-classes.

	1D CNN + inexact q-mer					lncLocator			
	Nucleus	Cytoplasm	Ribosome	Exosome	Cytosol	Nucleus	Cytoplasm	Ribosome	Exosome
Sn	0.748±0.106	0.885±0.047	0.767±0.161	0.34±0.237	0.3±0.221	0.3815	0.8801	0.07	0.04
Sp	0.928±0.042	0.719±0.087	0.975±0.02	0.991±0.012	0.958±0.031	0.9217	0.3636	0.9753	0.9727
MCC	0.692±0.054	0.615±0.055	0.723±0.13	0.403±0.224	0.292±0.214	0.357	0.288	0.07	0.015
OA	0.734±0.024					66.50%			

Table 15. A comparison between the proposed method and lncLocator. The data is extracted from iLoc-lncRNA [19].

Table 16 shows a performance comparison between the proposed method in this work and the iLoc-lncRNA. The proposed method uses 8-mer with 7 mismatches. The average accuracy of our approach is 92.1%±2.4%, which is 5.38% higher than iLoc-lncRNA.

	1D CNN + inexact q-mer				iLoc			
	Nucleus	Cytoplasm	Ribosome	Exosome	Nucleus	Cytoplasm	Ribosome	Exosome
Sn	0.884±0.048	0.962±0.02	0.922±0.071	0.517±0.229	0.7756	0.9906	0.4651	0.1667
Sp	0.975±0.012	0.861±0.038	0.992±0.01	0.996±0.005	0.9759	0.6768	0.9983	1
MCC	0.87±0.046	0.839±0.042	0.905±0.066	0.622±0.243	0.796	0.742	0.652	0.4
OA	92.1%±2.4%				86.72%			

Table 16. A comparison between the proposed method and iLoc-lncRNA

From the tables and figures shown above, we can see a general tendency that the accuracy increases with q increases. The deep learning method did an excellent job on subcellular localization. In (q, k) mismatch model, we can improve the prediction accuracy. When we take 8-mer with 7 mismatches, we can get the overall accuracy, an average of 0.921±0.024 and the highest 0.947 on iLoc dataset. The inexact q-mer may add more crucial biological information to the segments. And this can help to recognize the location where lncRNA resides in the cell.

4.5 The impact of length of lncRNA sequence

I checked prediction accuracy based on the lncRNA sequence lengths. There are 131 lncRNA sequences in the iLoc test set and 105 in the lncLocator dataset. Table 17 shows an average prediction accuracy, of 10 runs, of different length lncRNA sequences on the lncLocator dataset with 8-mer. The "Correct" in the table means the predicted class is the same as the actual class; incorrect otherwise. When the length of lncRNA sequences is between 2000 and 3000, the prediction accuracy is the highest, 76.7%. Figure 9 shows the performance associated with the length of the sequence. Table 18 shows, when using 8 mer, the group of length between 2000 and 3000 can get the best performance, 95.01%, on the iLoc dataset. Figure 10 visualizes the performance associated with the length of the lncRNA sequence. From the results on the two datasets, we can conclude that when the lncRNA sequence length is between 2000 and 3000, the sequence will provide more information for the prediction on lncRNA subcellular localization on the two datasets.

lncLocator	< 1000	1000~2000	2000~3000	> 3000
Correct	16.8	26.3	21.7	23
Incorrect	10.4	9.9	6.9	8
Total	27.2	36.2	28.6	31
Accuracy %	61.74	73.28	76.7	74.2

Table 17. The prediction accuracy associated with length of lncRNA sequence on lncLocator dataset. Correct See text.

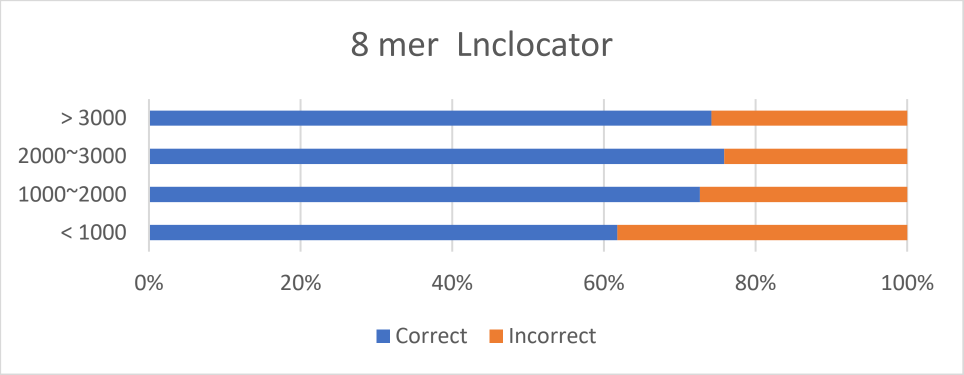


Figure 9. The performance associated with the length of lncRNA sequence on lncLocator dataset.

iLoc	<1000	1000~2000	2000~3000	>3000
Correct	24.2	39.1	27.6	26.1
Incorrect	6.2	4.5	1.5	1.8
Total	30.4	43.6	29.1	27.9
Accuracy %	79.85	89.67	95.01	93.79

Table 18.The prediction accuracy associated with the length of lncRNA sequence on iLoc dataset.

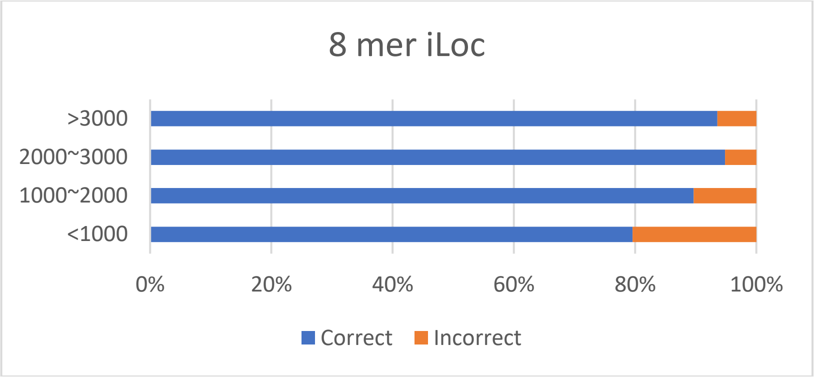


Figure 10.The performance associated with the length of lncRNA sequence on iLoc dataset.

Chapter 5: Conclusion & Discussion

LncRNAs can exist in different parts of the cell and show some crucial biological functions that may cause diseases. Therefore, understanding their subcellular localization becomes an urgent task. However, compared to the vast lncRNA family, people have annotated very few of them with their subcellular localization. The property of the lncRNA sequence makes it possible to annotate the lncRNAs subcellular localization using computational methods based on the existing lncRNA atlas. The conventional computational methods annotate the lncRNA subcellular localization by extracting q-mer profiles from the lncRNA sequence. Then, they train Machine Learning models with q-mer profiles and get impressive results. Given the gene mutation, there may be some changes in the lncRNA sequence, and these changes exhibit various biological functions which can cause diseases. We hypothesize the changes may affect the subcellular localization.

In this paper, to test the hypothesis, we train a 1D CNN model with q-mer profile. To compare the performances, we try q-mer with various mismatches. The results show an upward trend in overall accuracy when the number of mismatches increased. It turns out that the mismatch on q-mer profile can improve the prediction performance. The proposed approach surpasses the state-of-the-art methods in predicting subcellular localization of lncRNAs.

The length of lncRNA sequence is proved to work on predicting the subcellular localization. When the sequence length is between 2000 and 3000 nucleotides, our model can get the best performance than other groups. Given the datasets are relatively small, this conclusion need to be justified in future work.

We acknowledge some potential limitations in this work. First, the datasets used are relatively small. Only hundreds of lncRNAs are contained in these datasets. It is hard to extract sufficient information from a small dataset to predict new unannotated lncRNAs. Second, the dataset is unbalanced. With unbalanced datasets, a model may perform well at predicting the majority classes while doing poorly in minority classes. More specific attention to this data imbalance problem could improve the results further. Third, the length of the lncRNA sequences varied from hundreds to thousands which may cause a significant difference in the sparsity of the feature map. How to extract a helpful *q*-mer profile from this potentially sparse feature space could pose a significant challenge. Fourth, the CNN model is flexible. We can easily add the different modules to the network, but getting the optimal model with appropriate hyperparameter tuning is still a key challenge in deep learning. Finally, as acknowledged earlier, computational challenges abound concerning time and space with the potential exponential increase in the feature space as q increases. These issues make a strong case for possible feature direction using this idea of inexact q-grams, especially given the improved comparative performance over state of the art. Advanced data structures and algorithmic techniques could be brought to bear on the computational challenges.

References

[1] J. Brosius, “The fragmented gene,” *Ann N Y Acad Sci*, vol. 1178, pp. 186–193, Oct. 2009, doi: 10.1111/j.1749-6632.2009.05004.x.

[2] C. Han Li and Y. Chen, “Small and Long Non-Coding RNAs: Novel Targets in Perspective Cancer Therapy,” *Current Genomics*, vol. 16, no. 5, pp. 319–326, Oct. 2015, doi: 10.2174/1389202916666150707155851.

[3] L. Ma *et al.*, “LncBook: a curated knowledgebase of human long non-coding RNAs,” *Nucleic Acids Research*, vol. 47, no. D1, pp. D128–D134, 2019, doi: 10.1093/nar/gky960.

[4] G. H. Reference, *What is a gene?* [Online]. Available: https://ghr.nlm.nih.gov/primer/basics/gene

[5] A. F. Palazzo and E. S. Lee, “Non-coding RNA: what is functional and what is junk?,” *Frontiers in Genetics*, vol. 6, p. 2, 2015, doi: 10.3389/fgene.2015.00002.

[6] J. A. Reuter, D. V. Spacek, and M. P. Snyder, “High-throughput sequencing technologies,” *Molecular Cell*, vol. 58, no. 4, pp. 586–597, May 2015, doi: 10.1016/j.molcel.2015.05.004.

[7] J. Zhu, H. Fu, Y. Wu, and X. Zheng, “Function of lncRNAs and approaches to lncRNA-protein interactions,” *Sci China Life Sci*, vol. 56, no. 10, pp. 876–885, Oct. 2013, doi: 10.1007/s11427-013-4553-6.

[8] L. Ma, V. B. Bajic, and Z. Zhang, “On the classification of long non-coding RNAs,” *RNA biology*, vol. 10, no. 6, pp. 925–933, Jun. 2013, doi: 10.4161/rna.24604.

[9] Y. Fang and M. J. Fullwood, “Roles, Functions, and Mechanisms of Long Non-coding RNAs in Cancer,” *Genomics, Proteomics & Bioinformatics*, vol. 14, no. 1, pp. 42–54, Feb. 2016, doi: 10.1016/j.gpb.2015.09.006.

[10] N. N. Parikshak *et al.*, “Genome-wide changes in lncRNA, splicing, and regional gene expression patterns in autism,” *Nature*, vol. 540, no. 7633, pp. 423–427, 2016, doi: 10.1038/nature20612.

[11] Q. Luo and Y. Chen, “Long noncoding RNAs and Alzheimer’s disease,” *Clinical Interventions in Aging*, vol. 11, pp. 867–872, 2016, doi: 10.2147/CIA.S107037.

[12] Z. Bao, Z. Yang, Z. Huang, Y. Zhou, Q. Cui, and D. Dong, “LncRNADisease 2.0: an updated database of long non-coding RNA-associated diseases,” *Nucleic Acids Research*, vol. 47, no. D1, pp. D1034–D1037, 2019, doi: 10.1093/nar/gky905.

[13] L.-L. Chen, “Linking Long Noncoding RNA Localization and Function,” *Trends in Biochemical Sciences*, vol. 41, no. 9, pp. 761–772, Sep. 2016, doi: 10.1016/j.tibs.2016.07.003.

[14] J. Carlevaro-Fita and R. Johnson, “Global Positioning System: Understanding Long Noncoding RNAs through Subcellular Localization,” *Molecular Cell*, vol. 73, no. 5, pp. 869–883, 2019, doi: 10.1016/j.molcel.2019.02.008.

[15] T. Cui *et al.*, “RNALocate v2.0: an updated resource for RNA subcellular localization with increased coverage and annotation,” *Nucleic Acids Research*, no. gkab825, Sep. 2021, doi: 10.1093/nar/gkab825.

[16] D. Mas-Ponte, J. Carlevaro-Fita, E. Palumbo, T. Hermoso Pulido, R. Guigo, and R. Johnson, “LncATLAS database for subcellular localization of long noncoding RNAs,” *RNA (New York, N.Y.)*, vol. 23, no. 7, pp. 1080–1087, 2017, doi: 10.1261/rna.060814.117.

[17] M. N. Cabili *et al.*, “Localization and abundance analysis of human lncRNAs at single-cell and single-molecule resolution,” *Genome Biology*, vol. 16, p. 20, Jan. 2015, doi: 10.1186/s13059-015-0586-4.

[18] 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 (Oxford, England)*, vol. 34, no. 13, pp. 2185–2194, 2018, doi: 10.1093/bioinformatics/bty085.

[19] Z.-D. Su *et al.*, “iLoc-lncRNA: predict the subcellular location of lncRNAs by incorporating octamer composition into general PseKNC,” *Bioinformatics (Oxford, England)*, vol. 34, no. 24, pp. 4196–4204, 2018, doi: 10.1093/bioinformatics/bty508.

[20] B. L. Gudenas and L. Wang, “Prediction of LncRNA Subcellular Localization with Deep Learning from Sequence Features,” *Scientific Reports*, vol. 8, no. 1, p. 16385, 2018, doi: 10.1038/s41598-018-34708-w.

[21] R. Karki, D. Pandya, R. C. Elston, and C. Ferlini, “Defining ‘mutation’ and ‘polymorphism’ in the era of personal genomics,” *BMC Med Genomics*, vol. 8, p. 37, Jul. 2015, doi: 10.1186/s12920-015-0115-z.

[22] S. Pennington, *Introduction to Genetics: 11th Hour*. Hoboken, UNITED KINGDOM: John Wiley & Sons, Incorporated, 1999. Accessed: Oct. 31, 2020. [Online]. Available: http://ebookcentral.proquest.com/lib/wvu/detail.action?docID=454434

[23] K. Waters, "Molecular Genetics," in *The Stanford Encyclopedia of Philosophy*, Fall 2013., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2013. Accessed: Oct. 26, 2020. [Online]. Available: https://plato.stanford.edu/archives/fall2013/entries/molecular-genetics/

[24] D. S. T. Nicholl, *An Introduction to Genetic Engineering*. Cambridge, UNITED KINGDOM: Cambridge University Press, 2008. Accessed: Oct. 31, 2020. [Online]. Available: http://ebookcentral.proquest.com/lib/wvu/detail.action?docID=343497

[25] J. X. Yang, R. H. Rastetter, and D. Wilhelm, "Non-coding RNAs: An Introduction," in *Non-coding RNA and the Reproductive System*, D. Wilhelm and P. Bernard, Eds. Dordrecht: Springer Netherlands, 2016, pp. 13–32. doi: 10.1007/978-94-017-7417-8_2.

[26] H. Lodish, A. Berk, S. L. Zipursky, P. Matsudaira, D. Baltimore, and J. Darnell, *Molecular Cell Biology*, 4th ed. W. H. Freeman, 2000.

[27] MedlinePlus Genetics, "What is noncoding DNA?" Retrieved from https://medlineplus.gov/genetics/understanding/basics/noncodingdna/ (accessed Nov. 02, 2020).

[28] J. Barciszewski, *Non-Coding RNAs: Molecular Biology and Molecular Medicine*. Springer Science & Business Media, 2003.

[29] A. Pask, "The Reproductive System," in *Non-coding RNA and the Reproductive System*, D. Wilhelm and P. Bernard, Eds. Dordrecht: Springer Netherlands, 2016, pp. 1–12. doi: 10.1007/978-94-017-7417-8_1.

[30] E. Alessio, R. S. Bonadio, L. Buson, F. Chemello, and S. Cagnin, "A Single Cell but Many Different Transcripts: A Journey into the World of Long Non-Coding RNAs," *Int J Mol Sci*, vol. 21, no. 1, Jan. 2020, doi: 10.3390/ijms21010302.

[31] E. Pennisi, "ENCODE Project Writes Eulogy for Junk DNA," *Science*, vol. 337, no. 6099, pp. 1159–1161, Sep. 2012, doi: 10.1126/science.337.6099.1159.

[32] J. T. Lee, “Epigenetic Regulation by Long Noncoding RNAs,” *Science*, vol. 338, no. 6113, pp. 1435–1439, Dec. 2012, doi: 10.1126/science.1231776.

[33] C. Charon, A. B. Moreno, F. Bardou, and M. Crespi, “Non-Protein-Coding RNAs and their Interacting RNA-Binding Proteins in the Plant Cell Nucleus,” *Molecular Plant*, vol. 3, no. 4, pp. 729–739, Jul. 2010, doi: 10.1093/mp/ssq037.

[34] J. Baek, B. Lee, S. Kwon, and S. Yoon, “LncRNAnet: long non-coding RNA identification using deep learning,” *Bioinformatics*, vol. 34, no. 22, pp. 3889–3897, Nov. 2018, doi: 10.1093/bioinformatics/bty418.

[35] R. Achawanantakun, J. Chen, Y. Sun, and Y. Zhang, “LncRNA-ID: Long non-coding RNA IDentification using balanced random forests,” *Bioinformatics*, vol. 31, no. 24, pp. 3897–3905, Dec. 2015, doi: 10.1093/bioinformatics/btv480.

[36] C. Yang *et al.*, “LncADeep: an ab initio lncRNA identification and functional annotation tool based on deep learning,” *Bioinformatics*, vol. 34, no. 22, pp. 3825–3834, Nov. 2018, doi: 10.1093/bioinformatics/bty428.

[37] U. K. Muppirala, V. G. Honavar, and D. Dobbs, “Predicting RNA-Protein Interactions Using Only Sequence Information,” *BMC Bioinformatics*, vol. 12, no. 1, p. 489, Dec. 2011, doi: 10.1186/1471-2105-12-489.

[38] L. Hu, Z. Xu, B. Hu, and Z. J. Lu, “COME: a robust coding potential calculation tool for lncRNA identification and characterization based on multiple features,” *Nucleic Acids Res*, vol. 45, no. 1, pp. e2–e2, Jan. 2017, doi: 10.1093/nar/gkw798.

[39] P. Dönnes and A. Höglund, “Predicting protein subcellular localization: past, present, and future,” *Genomics, Proteomics & Bioinformatics*, vol. 2, no. 4, pp. 209–215, Nov. 2004, doi: 10.1016/s1672-0229(04)02027-3.

[40] S. Clancy, “RNA functions,” vol. 1(1), p. 102, 2008.

[41] P. M. Macdonald, “mRNA localization: assembly of transport complexes and their incorporation into particles,” *Curr Opin Genet Dev*, vol. 21, no. 4, pp. 407–413, Aug. 2011, doi: 10.1016/j.gde.2011.04.005.

[42] J. Sprenger, J. Lynn Fink, S. Karunaratne, K. Hanson, N. A. Hamilton, and R. D. Teasdale, “LOCATE: a mammalian protein subcellular localization database,” *Nucleic Acids Research*, vol. 36, no. Database issue, pp. D230-233, Jan. 2008, doi: 10.1093/nar/gkm950.

[43] J. Seiler, M. Breinig, M. Caudron-Herger, M. Polycarpou-Schwarz, M. Boutros, and S. Diederichs, “The lncRNA VELUCT strongly regulates viability of lung cancer cells despite its extremely low abundance,” *Nucleic Acids Res*, vol. 45, no. 9, pp. 5458–5469, May 2017, doi: 10.1093/nar/gkx076.

[44] W. Chen, T.-Y. Lei, D.-C. Jin, H. Lin, and K.-C. Chou, “PseKNC: a flexible web server for generating pseudo K-tuple nucleotide composition,” *Analytical Biochemistry*, vol. 456, pp. 53–60, Jul. 2014, doi: 10.1016/j.ab.2014.04.001.

[45] J. M. Kirk *et al.*, “Functional classification of long non-coding RNAs by k-mer content,” *Nature Genetics*, vol. 50, no. 10, pp. 1474–1482, 2018, doi: 10.1038/s41588-018-0207-8.

[46] T. Derrien *et al.*, “The GENCODE v7 catalog of human long noncoding RNAs: analysis of their gene structure, evolution, and expression,” *Genome Research*, vol. 22, no. 9, pp. 1775–1789, Sep. 2012, doi: 10.1101/gr.132159.111.

[47] T. Zhang *et al.*, “RNALocate: a resource for RNA subcellular localizations,” *Nucleic Acids Research*, vol. 45, no. D1, pp. D135–D138, 2017, doi: 10.1093/nar/gkw728.

[48] Y. Fan, M. Chen, and Q. Zhu, “lncLocPred: Predicting LncRNA Subcellular Localization Using Multiple Sequence Feature Information,” *IEEE Access*, vol. 8, pp. 124702–124711, 2020, doi: 10.1109/ACCESS.2020.3007317.

[49] A. Ahmad, H. Lin, and S. Shatabda, “Locate-R: Subcellular localization of long non-coding RNAs using nucleotide compositions,” *Genomics*, vol. 112, no. 3, pp. 2583–2589, May 2020, doi: 10.1016/j.ygeno.2020.02.011.

[50] S. Zhang and H. Qiao, “KD-KLNMF: Identification of lncRNAs subcellular localization with multiple features and nonnegative matrix factorization,” *Analytical Biochemistry*, vol. 610, p. 113995, Dec. 2020, doi: 10.1016/j.ab.2020.113995.

[51] C. Leslie, J. Weston, E. Eskin, and W. S. Noble, “Mismatch String Kernels for SVM Protein Classification,” p. 8.

[52] C. Leslie, E. Eskin, and W. S. Noble, “THE SPECTRUM KERNEL: A STRING KERNEL FOR SVM PROTEIN CLASSIFICATION,” in *Biocomputing 2002*, Kauai, Hawaii, USA, Dec. 2001, pp. 564–575. doi: 10.1142/9789812799623_0053.

[53] D. Adjeroh, T. Bell, and A. Mukherjee, *The Burrows-Wheeler Transform: Data Compression, Suffix Arrays, and Pattern Matching*. 2008, p. 351. doi: 10.1007/978-0-387-78909-5.

[54] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[55] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, no. 85, pp. 2825–2830, 2011.

[56] R. S. Srinivasamurthy, “Understanding 1D Convolutional Neural Networks Using Multiclass Time-Varying Signals,” *undefined*, 2018, Accessed: Aug. 30, 2021. [Online]. Available: https://www.semanticscholar.org/paper/Understanding-1D-Convolutional-Neural-Networks-Srinivasamurthy/0416eb821e664b48a3f33e77566006aca3a3c5a3

[57] L. Prechelt, “Early Stopping - But When?,” in *Neural Networks: Tricks of the Trade*, G. B. Orr and K.-R. Müller, Eds. Berlin, Heidelberg: Springer, 1998, pp. 55–69. doi: 10.1007/3-540-49430-8_3.

[58] T. O’Malley *et al.*, “KerasTuner.” 2019. [Online]. Available: https://github.com/keras-team/keras-tuner

[59] S. Scardapane and D. Wang, “Randomness in neural networks: an overview,” *WIREs Data Mining and Knowledge Discovery*, vol. 7, no. 2, p. e1200, 2017, doi: 10.1002/widm.1200.