Alma Mater Studiorum – Università di Bologna

DOTTORATO DI RICERCA IN

Biologia Cellulare e Molecolare

Ciclo XXXI

Settore Concorsuale: 05/E2

Settore Scientifico Disciplinare: BIO/11

RNA syntax and semantics: investigating the transcriptome complexity

Presentata da: Bonafede Irene

Coordinatore Dottorato
Prof. Capranico Giovanni

Supervisore
Prof. Ferrè Fabrizio

Esame finale anno 2019

Table of Contents

Abstract

A large portion of eukaryotic genome is transcribed into RNAs that apparently have not a coding potential. The major part of these noncoding RNAs includes transcripts with size greater than 200 nucleotides, formally known as long noncoding RNAs (lncRNAs). In recent years, the number of publications dealing with this interesting class has steadfastly grown. In fact, they are emerging as key players in a wide range of cellular processes, including epigenetic modification, chromatin modulation, transcription, splicing and translation.

LncRNAs have cell type or tissue specific expression and, in contrast to other types of RNAs, they can localize both in the cytoplasm and nucleus, or, more rarely, in other subcellular compartments, which has recently increased the interest in conducting experiments, building databases and making available localization data for subsequent studies. Furthermore, lncRNAs generally lack primary sequence conservation, can be spliced, polyadenylated or not or even polymorphic, and possess the ability to adopt a secondary or tertiary structure that may influence the biological function. Moreover, research over recent decades has shown that RNA–protein interactions form a highly complex network involving numerous RNAs and proteins, and high throughput experiments to identify RNA-protein interactions are beginning to provide a large amount of valuable information.

The basic idea of this work is to reconstruct an heterogeneous network depicting lncRNA-protein interactions that would summarize what is currently known, allow the prediction of lacking features and thus give a complete mechanistic understanding of the functions of lncRNAs by the network topological analysis.

Unfortunately, this approach raised problems related to different aspects. Firstly, even if recent studies show that a growing number of lncRNAs play critical roles in complex cellular processes and that they are implicated in a wide range of human diseases, the fraction of annotated lncRNAs is still small.

Secondly, as of today, most databases are highly inhomogeneous in terms of the type of the provided information, and analytical and experimental approaches to investigate them have been hampered by the lack of comprehensive annotation.

Thirdly, the standard bioinformatics solution to fill the gaps due to lacking information is based on machine learning techniques that usually lead to myriad problems related to the

preprocessing of data and the input dataset format, both aspects that oftentimes are conducted by trial and error.

Finally, a challenging problem that arises in this domain is the data visualization. A common strategy used to overcome the problem is constructing interaction networks, whose analytical but also visual inspection can offer important biological insights, however one primary drawback with this approach is to develop an efficient and scalable algorithm to produce easily interpretable layouts for sparse graphs when the number of nodes is very large.

The thesis deals with a multidisciplinary approach to unravel the complexity of lncRNAs regulatory networks and investigate their functions. The objective is to demonstrate the feasibility of using machine learning techniques as well as network analysis to find hidden patterns in the data and to predict new features.

List of Acronyms

lncRBP	Long noncoding RNA binding protein
miRNA	microRNA
mRNA	messenger RNA
piRNA	PicoRNA
RBP	RNA Binding Protein
RF	Random Forest
rRNA	Ribosomial RNA
snoRNA	Small nuclear RNA
SSE	Secondary Structure Element
SVM	Support Vector Machine
tRNA	Transfer RNA
NLP	Natural Language Processing
NGS	Next Generation Sequencing
DMS	Dimethyl sulfate
SHAPE	Selective 2'-Hydroxil Acetylation by Primer Extension
CLASH	Cross-linking, ligation, and sequencing of hybrids

AMT	4′-aminomethyltrioxalen
SPLASH	Sequencing of psoralen-cross-linked, ligated, and selected hybrids
PARIS	Psoralen analysis of RNA interactions and structures
LIGR-Seq	ligation of interacting RNA followed by high-throughput sequencing
icSHAPE	in vivo click-selective 2′-hydroxyl acylation and profiling experiment
Fraq-Seq	Fragmentation sequencing
FISSEQ	fluorescent *in situ* sequencing
ChIRP	the Chromatin Isolation by RNA Purification
SubRNAseq	Subcellular RNA sequencing (SubRNAseq)
RAP	RNA Antisense Purification
RIP	Rna Immunoprecipitation
CHART	Capture Hybridization Analysis of RNA Targets
iCLIP	individual-nucleotide resolution
PAR-CLIP	photoactivable ribonucleoside-enhanced cross-linking and immunoprecipitation

List of Figures

List of Tables

1. Introduction

Noncoding RNAs, once thought as a part of transcriptional noise, are now emerging as central players controlling several cellular mechanisms. The noncoding RNAs have been classified based on their sequence length into small noncoding RNAs (<200 nucleotides) and long noncoding RNAs (>200 nucleotides). The latter represents the largest class of the mammalian transcriptome and this thesis considers it as the main subject of its study. According to the last version of NONCODE V5.0, there are about 172,216 distinct ncRNA transcripts in Human [1][2] and more than 500,000 including 17 species. Depending on their position and direction of transcription with respect to protein coding genes, lncRNAs may be classified as stand-alone or intergenic (distinct transcription units located in sequence space that don't overlap to protein coding genes), antisense (transcribed from the antisense DNA strand of annotated transcription units), long intronic (encoded within the introns of annotated genes), processed pseudogenes (replica of genes that have lost their coding capacity due to mutations, but that can still be transcribed), and promoter/enhancer associated transcripts (transcribed in correspondence to these DNA units and generally associated to their functions). They may regulate genes in close proximity (in *cis)* or at a distance (in *trans*) from their transcription site. The majority of lncRNAs have not been functionally characterized, but those for which information is available, are reported to play important and varied roles in cellular processes (e.g. maintaining homeostasis, regulating cell growth and differentiation, apoptosis, imprinting, promoting pluripotency and controlling gene expression), suggesting the hypothesis that they represent an important layer of regulators inside the cell. Broadly speaking, the biological functions of lncRNAs include translation of genetic information, cellular signal transduction and transcriptional regulation.

The association of RNAs with other nucleic acids (DNA and RNA) and with proteins is of paramount importance for understanding cell growth, development and differentiation, evolution and disease[3]. In particular, it has been shown that RNA and proteins interactions are involved in many cellular processes and can imply either transient or stable nucleoprotein complexes encompassing specific and non-specific interactions. Not surprisingly, a vast number of works have provided deep insights into the functional implication of RNA-binding complexes features in terms of sequences and structures. In addition, the mechanism of action may be diverse. lncRNAs may bring a group of proteins into spatial proximity

acting as scaffolds, or may recruit a protein or a complex to DNA acting as guides, or bind and titrate away a protein target without exerting any additional functions and acting as competitors, or finally acting as enhancers involved in chromosomal looping [4]. Moreover, genomic and transcriptomic studies of the primary sequence conservation of protein-coding and non-coding loci revealed that the human genome is highly diverse, particularly for its non-coding fraction. It is reported that only 2.2% of its DNA sequence is subjected to conservation constraints [3] and non-coding genes are the least conserved [2]. However, although the body of non-coding genes tends not to be conserved, there are other criteria to look at in order to extract useful information. Remarkably, different studies have demonstrated the existence of short peaks of conserved sequences in specific portions of a gene, such as the 5' ends [4]. In addition, lncRNAs may or not be 3'polyadenylated or, to add complexity, they may present both forms, like NEAT1 or MALAT1, as polymorphic transcripts. Furthermore, the cellular localization of a lncRNA is informative regarding its function. For example, nuclear lncRNAs could plausibly have functions in histone modification or direct transcriptional regulation, while cytoplasmic lncRNAs were found linked to mono or polyribosomal complexes, even though this association is not clear. Some studies suggest a role in translation, while other in lncRNAs decay. In any case, the possibility that lncRNAs contain a short open reading frame that is translated should also be considered [4]. Hence, there is a clear need to understand how these molecules and the interactions in which they are involved determine the function of this complex machinery, and a major challenge of contemporary biology is to embark on an integrated theoretical and experimental program to map out, understand and model in quantifiable terms the topological and dynamical properties of this enormous class of transcripts.

Databases as RAIN [5], lncRNAdb [6], RNAlocate [7], LNCipedia [8], NONCODE [2], RNAcentral [9], Ensembl [10] provide a huge number of data about the biological roles and characteristics of single lncRNAs. On the other hand, RAIN [5] and NPInter [11] are the only databases which provide ncRNA-RNA and ncRNA-proteins and protein-protein interactions. These ncRNA-protein associations have been established from curated examples, experimental data, interaction predictions and automatic literature mining. The key problem is to develop a systematic approach to data analysis in order to understand the single interactions as well as how the sum of these interactions can affect or guide the cell's behavior. One approach to solve this problem involves the use of system biology that aims at understanding biological molecules not only as individual entities but as interacting

systems. In particular, taking advantage of rules provided by the field of graph theory, it is possible to build and analyze graphs depicting interaction networks. The key idea is to represent different entities as nodes and link them by edges that convey information about the nodes interactions. Depending on the nature of the underlying edge information, different types of analyses can be performed. Moreover, the interactions can be undirected, in which the relationship is a simple connection without an implied given flow (e.g. protein-protein interaction networks or miRNA-lncRNA interaction network) or directed, in which a clear flow is implied (e.g. metabolic or cell signaling networks). However, there is a further problem with the available information, due to the fact that the databases usually use different identifiers and this leads to the well-known problem in looking for homogeneous features. For these reasons, it would be valuable to develop prediction methods based on an unambiguous and ubiquitous characteristic (e.g. the primary sequence) that can be used to identify potential partners in the absence of experimental features, which might be informative but not always, or even rarely available for all the considered molecules. These methods can then be used for modeling regulatory networks. The standard solution to the problem is based on machine learning techniques, which are algorithms able to learn patterns from provided examples without being specifically programmed. The typical pipeline includes:

1. Data preparation
2. Data representation
3. Prediction model
4. Downstream analysis

In general, the aim is to build a model that can be used to make predictions based on the available evidence, in the presence of uncertainty. Specifically, the learning algorithms identify patterns in the data, learning from the observations. When exposed to more observations, the machine improves the model, and in turn improves its predictive performance.

Machine learning algorithms typically require a numerical representation of data points to make them suitable for processing and statistical analysis. This numerical representation is usually in the form of a vector containing multiple elements describing each object, named the feature vector. A consequence is that for data that are not numerical in nature, such as a biological sequence, finding a suitable way to represent it is mandatory and often not trivial.

Remarkably, the feature vectors construction strictly depends on the choice of the prediction model. In fact, methods based on classical machine learning algorithms such as Support Vector Machine (SVM) or Random Forest (RF) require fixed-length feature vectors, independently on the data point size. This means that two biological sequences of markedly different length must be nevertheless encoded by vectors of the same size. More recently, deep learning approaches and above all text processing are changing the way to view and describe biological sequences. In particular, word embedding techniques based on deep learning have been proposed as a more advanced approach to process textual information. The key idea is to give the learning algorithm a text with associated labels and create a model. Once the model is obtained, it is possible to take new bits of text and include them into the model, finally obtaining as output the predicted classification for that text. Such methods were developed, and find their natural application, for the analysis of human languages. However, biological sequences are obviously not a real text, but if one can find a method to depict them as words and sentences, the concept is actually very simple: to learn everything we can learn by machine learning algorithms and look for rules along the sequences. Furthermore, one can go beyond the primary sequence and include in the model additional features that could improve its accuracy. For example, while earlier approaches for the rationalization of protein-RNA interaction determinants were focused on the use of sequence information only, recent studies [12–14] suggested that the RNA secondary structure and the secondary structure elements size have a role in the interaction process. In fact, it seems that elements with different length or sizes can have different functional roles. As a consequence, including secondary structure elements in the training features could result in the improvement of the model.

Although this research area is rapidly expanding, there are some difficulties to overcome. Among them the representation of biomolecules and the difficulty in establishing an interacting and non-interacting dataset (since negative examples are often required for the model training), and it is also unclear whether the available experimental data are sufficient for successfully training classifiers.

To illuminate this uncharted area, our aim is to provide a generic strategy that includes breaking the sequence into individual units (words), treat them as a text and use machine learning techniques to predict missing information, in order to build a network to visualize and analyze biologically meaningful lncRNAs interactions. With this in mind, this chapter outlines the different available strategies to investigate the lncRNAs interactome, subcellular

localization and secondary structure, which are all key features of the learning strategy that we developed. The aim of Chapter 1 is also to give an overview about the origin of our data. We will show methods for comprehensive experimental identification of lncRNA-protein interactions, lncRNA subcellular localization and lncRNA secondary structure. In Chapter 2, we will focus on *in silico* analysis dealing with the questions that characterize the computational process: how convert a biological entity into a suitable input for computer algorithms, which model should be used and finally which information is more informative and how to visualize it. The results are presented and discussed in the Results and Discussion (Chapter 3). Chapter 4 focuses on conclusions and future perspectives while Chapter 5 outlines the material and methods employed in this study.

This section will focus on the lncRNAs investigation tools as described in Figure 1, which represents a rational classification of the available analysis approaches.

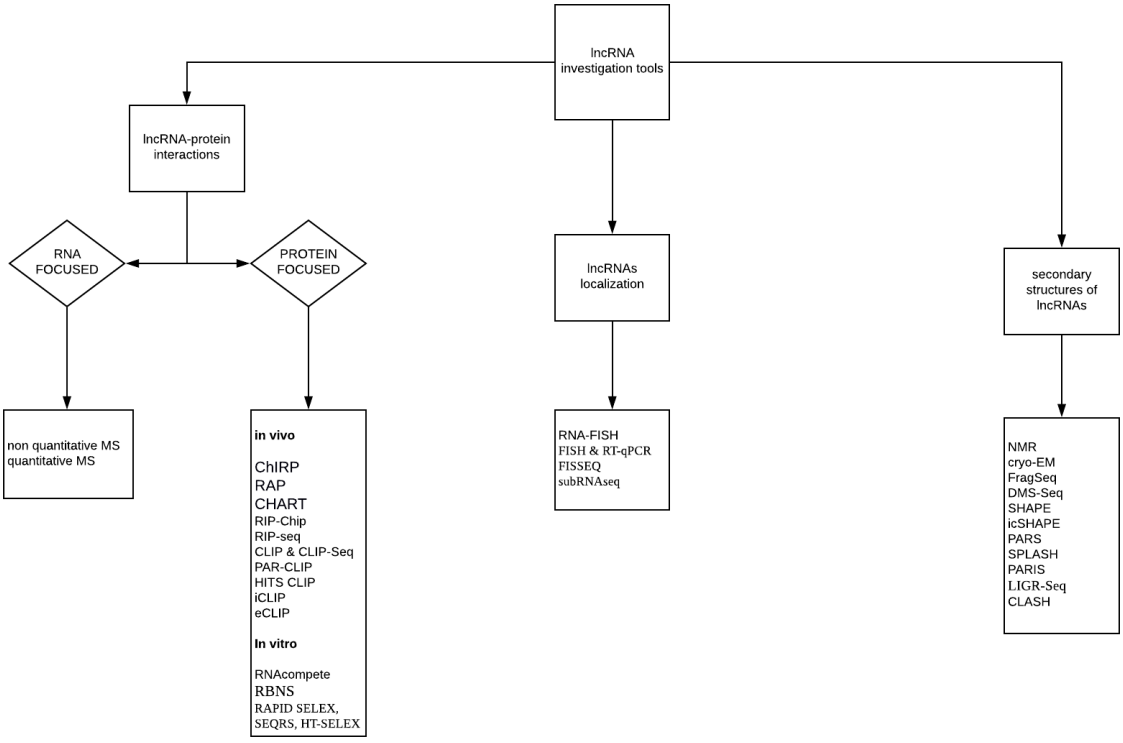


Figure 1: overview of different tools for lncRNA investigation

1.1. Investigating the lncRNA interactome

Research over recent decades has shown that RNA–protein interactions form a highly complex network involving numerous lncRNAs. Specific short sequences in the RNA sequence or larger secondary or tertiary structures are involved in these interactions. Moreover, lncRNAs length, subcellular localizations, genomic localization and expression level may be play a key role to the functional role of lncRNAs [15].

A key point is that lncRNAs is a very heterogeneous class. Many lncRNAs are reported to interact with one specific protein, other with multiple regulatory complexes simultaneously.

Some lncRNAs possess regulatory functions, while others are merely by-products of transcription. The length spans from 200 bp to 2,2 kbp for HOTAIR to several kbp for Kcnq1ot, a 91 kbp long noncoding RNA that maps to the protein coding Kcnq1 gene in antisense orientation [16].

For such long lncRNAs, by binding multiple effector partners at the same time by means of different domains, they would explicate the role of scaffolds and facilitate the interaction of their partners. Moreover, they can also act as inhibitors, for example by binding to specific transcription factors acting as decoys and preventing their association with DNA.

In addition, lncRNAs would direct the localization of ribonucleoprotein complexes to specific targets acting as guides. Some of them regulate the expression of genes in *cis* (on neighboring genes), remaining linked to their transcription sites and interacting with proteins, others can change the gene expression in *trans* (on distantly located genes). They also can bind to enhancers and help them in their activity (e.g. by promoting the formation of chromatin loops)

Finally, they show cell type specific expression, and the transcription of individual lncRNAs can occur at very specific times and places; hence, it has been suggested that they can serve as signals to integrate developmental cues, interpret cellular context, or respond to diverse stimuli.

All these features are required to be taken into account in order to better understand the lncRNAs interactome. In this area, given the complexity of the interactions and the large number of lncRNAs that a genome can express, *in silico* methods can be of primary importance for the characterization of lncRNAs.

The depiction of the central regulative role of RNA in general has been facilitated by technological advances, and different methods were developed in the past decades to uncover the interaction between proteins and RNAs (e.g. RIP, CLIP or its variants). The conventional methods were recently coupled with high throughput experiments to identify more systematically RNA-protein interactions, providing a large amount of valuable information about the complexity of the RNA-protein interaction networks, in turn requiring reliable computational methods for analyzing and organizing them.

The methods to uncover proteins and RNAs interactions can be classified in RNA-focused and protein-focused. The goal of RNA-focused approaches is the identification of all proteins bound to an RNA of interest. On the contrary, in protein-focused methods the goal is to identify RNAs bound by a protein of interest [17]. In general, the RNA-focused approach aim is often related to the identification of lncRNA-chromatin interactions (e.g. ChIRP or CHART) as well as to lncRNA-RNA interactions (RAP and CLASH), while the protein focused goal is to determine lncRNA bound to a protein of interest.

In the next subsections we will briefly describe the main methods for the detection of the lncRNAs interactome giving a quick account on the RNA-focused methods that are not part of the data employed in this thesis, and giving more attention to the protein-focused methods that were chosen as starting point for our work.

1.1.1. RNA-focused methods

While the protein-focused methods use an antibody to capture a protein of interest and sequencing the associated RNA, these methods purify an RNA of interest and identify the associated protein complexes.

The RNA focused methods can be divided in *in vivo* and *in vitro*. In *in vivo* methods, cross-linking between proteins and RNA is induced by UV or formaldehyde, allowing the stabilization of physiological interactions by covalent bonds. Then, cells are lysed, the RNA of interest is captured and bound proteins are detected.

The *in vitro* approaches are based on the immobilization of a synthetic RNA bait on a support dipped by a cell lysate or by a protein library to capture and identify proteins. The main difference between *in vivo* and *in vitro* methods is that the *in vivo* approaches preserve the context of true RNA-protein interactions but understandably, they are more technically challenging, especially if the target RNA is of low abundance in the cell [18]. After washing

and elution, proteins showing affinity towards the immobilized RNAs can be isolated and identified by mass spectrometry (MS).

MS2 trapping

One general approach to capture RNA is to exploit the naturally occurring interactions between RNA and protein - such as the bacteriophage MS2 viral coat protein, which binds tightly to an RNA stem-loop structure. This strategy employs a MS2 bacteriophage coat protein to specifically select a stem-loop structure of viral origin inserted at the 3' end of the lncRNA of interest. lncRNAs containing that hairpin bind to the coat protein, which in turn is covalently bound to a solid support. Bound RNA-protein complexes can then be washed, eluted and identified by MS.

SILAC

Stable isotope labelling with amino acids in cell culture (SILAC) is a simple approach for the *in vivo* incorporation of a detectable label into proteins. SILAC labels cellular proteomes through normal metabolic processes, incorporating non-radioactive, stable isotope-containing amino acids in newly synthesized proteins. Natural amino acids are replaced by SILAC amino acids. The former are lighter than the latter. Hence, when two cell populations (one labelled, the other not labelled) are mixed, their proteins remain distinguishable by MS because of the molecular weight difference.

The MS can be quantitative or non-quantitative. In the quantitative methods, the protein abundances are determined from the relative MS signal intensities obtained comparing the same proteins in the sample and in the control (e.g. applying SILAC and measuring the mass to charge ratio of ions to identify and quantify molecules). In the non-quantitative methods, purified proteins from the RNA sample of interest and a control are separated by gel electrophoresis and stained for total protein. Protein bands that are present only in the sample of interest but not the control are extracted and the proteins identified by MS. Alternatively, the total proteome can be analyzed by MS to detect all proteins purified in a sample [18].

1.1.2. RNA interaction with chromatin and RNA

ChIRP-RAP-CHART

In the Chromatin Isolation by RNA Purification (ChIRP) technique, macromolecular interactions are cross-linked with formaldehyde, nuclei isolated, lysed and sonicated. The fragments then passed through beads coated with streptavidin and bound by biotinylated DNA oligonucleotides antisense to a target RNA, so that the RNA is specifically recognized and hybridized. After washing and purification, the genomic regions bound to the target RNA can be identified by high-throughput DNA sequencing. Proteins associated with the target RNA can be analyzed by mass spectrometry or immunoblotting. The RNA Antisense Purification (RAP) and Capture Hybridization Analysis of RNA Targets (CHART) methods are similar to ChIRP, differing mostly in the design strategy of the antisense oligonucleotides and in the cross-linking protocols. While these methods are offering important evidence for the involvement of lncRNAs in gene expression regulation and chromatin remodeling, it should be noted that they cannot prove direct lncRNA–protein binding.

CLASH

The first high-throughput method using proximity ligation, termed cross-linking, ligation, and sequencing of hybrids (CLASH) [19], was developed to study *in vivo* RNA duplexes recognized by a specific RNA-binding protein (RBP). CLASH uses a modified version of the CLIP protocol, described in more detail later. Like CLIP, CLASH uses UV-C irradiation to cross-link RBPs to the bound RNAs, followed by immunopurification of the RBP–RNA complex. These RNA fragments are then identified using high-throughput DNA sequencing. In CLASH, there is an additional proximity ligation step that is designed to ligate together the two arms of the isolated RNA duplexes.

1.1.3. Protein-focused methods

As we did for the RNA-focused methods, we can broadly divide the protein-focused methods in two categories: *in vivo* and *in vitro* methods. *In vivo* methods (e.g. RIP-Chip, RIP-seq and the various CLIP strategies), are based on covalent bonds induced by UV between RNA nucleotides and proximal RBP amino acids at the binding sites. Protein and RNA complexes are then isolated by immunoprecipitation using an antibody specific for an RBP. In the *in vitro* approaches (SELEX, RNAcompete and RBNS), protein baits are immobilized to a

support and exposed to RNAs. After cycles of selections and amplifications, RNA is isolated and sequenced [20].

1.1.3.1. *In vivo* protein-focused

RIP

Rna Immunoprecipitation (RIP) is a protein immunoprecipitation in which the RBP of interest is immunoprecipitated together with its associated RNAs for identification of the bound transcripts; then, the detection of these bound transcripts is performed by microarrays (RIP-Chip) [21] or sequencing (RIP-seq) [22]

CLIP strategies

CLIP are a class of methods to identify RNA-Protein interaction by cross-linking cells using 254 nm ultraviolet radiation to covalently cross-link *in vivo* RNA-protein complexes. After cross-linking, the complex is immunoprecipitated, subjected to RNase treatment, followed by proteinase K digestion, 5'adaptor ligation and purification. The purified RNA fragments are adapter-ligated, amplified by PCR and sequenced. HITS-CLIP, individual-nucleotide resolution (iCLIP), photoactivable ribonucleoside-enhanced cross-linking and immunoprecipitation (PAR-CLIP) are variants of this general protocol, in combination with high-throughput techniques for detection. iCLIP provides information of the cross-link sites at nucleotide resolution. The proteinase K treatment digests the covalently bound proteins, leaving only the cross-linked amino acids. Then, unlike CLIP, RNAs undergo directly to reverse transcription without being subjected to 5'RNA adaptor ligation.

During the reverse transcription, amino acids bound to RNAs can cause the reverse transcriptase to detach, truncating prematurely the cDNAs at the cross-linking nucleotide. On the other hand, the unbound RNAs are converted into full-length cDNAs. Amplifying and comparing the two types of cDNAs, the protein binding sites are detected at single nucleotide resolution.

The PAR-CLIP method utilizes photoreactive ribonucleoside analogues, such as 4-thiouridine (4-SU) and 6thioguanosine (6-SG), which in turn allow the use of UV light of 365 nm in order to improve cross-linking efficiency. In addition, in response to cross-linking, specific sequence transitions like T to C in 4SU and G to A in 6SG are induced during the reverse transcription, which can be used to identify the precise position of cross-linking and to better discriminate between cross-linked RNAs and abundant cellular RNAs [20].

Recently, Enhanced CLIP (eCLIP) has caught the attention of the research community as a means of achieving better specificity and positional resolution. As in CLIP, eCLIP is based on the covalent link induced by UV irradiation, RNA fragmentation, immunoprecipitation of a targeted protein along with cross-linked RNA, and conversion of that RNA into double-stranded DNA high-throughput sequencing libraries through adapter ligation and reverse transcription. The two protocols are different in the addition of adapters: in iCLIP is a one-time step, while in eCLIP an indexed 3' adapter is ligated to the cross-linked RNA fragment while on the immunoprecipitation beads, and a 3' RNA adapter is ligated after reverse transcription [23]. It has been shown that this technique:

- Maintains the single-nucleotide resolution identification of RBP binding sites from previous methods
- Dramatically decreases the required amplification and greatly enhances the rate of success at generating libraries with high usable read percentages
- Allows the binding site identification with decreased sample requirements and high reproducibility for individual studies

To summarize, RIP, RIP-Chip and RIP-Seq allow the identification of bound transcripts, but do not provide direct information about the localization of the binding site, while CLIP strategies identify the binding sites with high (often single-nucleotide) resolution.

1.1.3.2. *In vitro* protein-focused

SELEX

The SELEX (Systematic Evolution of Ligands by EXponential enrichment) technology relies on the ability to separate RNAs having high affinity for a purified protein from a library of RNAs with random or semi-random sequence. SELEX experiments are performed over several cycles with each round resulting in increased enrichment of RNAs capable of binding to the protein. After PCR amplification, they are cloned and sequenced by the Sanger method. There are many variations of SELEX strategies, such as HT-SELEX [24], SEQRS [25] and RAPID-SELEX [26], but all are based on the ability to separate bound RNA from unbound RNA with the aim to identify a larger number of bound RNAs reducing the number of rounds.

RNAcompete and RNA Bind-n-Seq

RNAcompete involve the generation of an RNA pool comprising different short (seven or eight nucleotides long) RNA sequences and structures; a single pulldown of the RNAs bound to a tagged RBP of interest; and finally microarray and computational interrogation of the relative enrichment of each RNA in the bound fraction relative to the starting pool [27].

A variant is RNA Bind-n-Seq (RBNS) [28], in which RNAs from a random library are incubated at different concentrations with the purified RBP of interest. The RNA is then reverse-transcribed and deep-sequenced. These methods not only allow for the identification of the bound RNAs, but can also be used to estimate binding affinity.

1.2. Investigating lncRNAs localization

Many fundamental characteristics of lncRNAs, such as absolute abundance and subcellular localization, remain unclear. In fact, lncRNAs can accumulate to specific nuclear bodies or they can be exported to the cytoplasm to exert their functions. In the cytoplasm they can act by sequestering a protein or interfering with protein post-translational modifications [3]. lncRNAs can be exclusively cytosolic (e.g. DANCR and OIP5-AS1), nuclear (e.g. NEAT1) or have a dual localization (HOTAIR) [29]. A small number of lncRNAs have also be detected in other subcellular compartments.

Furthermore, the nucleus is highly organized and compartmentalized containing several different nuclear bodies such as the nucleoli, nuclear speckles and nuclear associated paraspeckles. All of them are characterized by the presence of specific lncRNAs and proteins and, of note, they lack a well-defined membrane separating them form their surroundings. Nevertheless, they are structurally distinct. In order to maintain the genetic material within a very small nuclear volume, the genomic DNA is highly packaged but maintaining the plasticity needed for efficient readout, processing and transfer of genetic information.

In general, the subcellular localization and the cellular distribution, in combination with the detection of the interaction partners and the expression levels can shed light on the lncRNAs functions. For example, lncRNAs associated with specific sub-nuclear domains and that co-localize with specific proteins like NEAT1, localized in paraspeckles together with paraspeckle proteins, are likely to have a structural role [30] as well as chromatin associated lncRNAs are more likely to have a regulatory role. For this reason, there is the necessity to analyze data coming from several experimental methodologies to unravel the potentially

regulatory functions of lncRNAs. Hereinafter we will describe the most important approaches currently used.

FISH

Fluorescence *in situ* hybridization FISH (or single molecule counting FISH) and qRTPCR or a combination of them are probably the gold standards for the detection of RNA in single cells. In particular, these methods were successfully used to define the subcellular localization of some well-known and functionally characterized lncRNA such as NEAT1, NEAT2, MALAT1 [31] and XIST [32], all localized in the nucleus.

The idea of FISH is that nucleic acids with complementary sequences tend to form a double helix. It can be DNA:DNA, RNA:RNA or RNA:DNA. It is a methodology that utilizes fluorescent nucleic acid probes that are complementary to target RNA sequences within the cell. After the probes hybridization to their targets, it is possible to detect them via fluorescence microscopy. It can in principle yield absolute counts of molecules at subcellular resolution. Another technique is the reverse transcription (RT) followed by conventional or quantitative (q) polymerase chain reaction (qPCR). The main steps are RNA isolation, reverse transcription to convert RNA template into complementary (cDNA), followed by a PCR amplification and quantification. In traditional RT-PCR the presence of the product is checked at the end of the reaction, while in RT-qPCR the amplification is tested at the end of every cycle. Often the FISH and RT-qPCR are combined in order to overcome each the limitations one of the other. In fact, RNA-FISH provides specific information about RNA localization within a cell population or a tissue, and RT-qPCR complements those results by giving an absolute measurements of transcript numbers [33].

FISSEQ

In 2015 Lee et co-workers introduced fluorescent *in situ* sequencing, FISSEQ [34], even though at present only few datasets are available, including several hundreds of lncRNAs. FISSEQ converts endogenous RNA molecules into short cDNA fragments *in situ* using random hexamer-primed reverse transcription (RT). The cDNA fragments are circularized and amplified using rolling circle amplification (RCA) *in situ,* followed by *in situ* next-generation sequencing (NGS) reactions. In the end, FISSEQ generates 3D images containing NGS reads at each pixel for data analysis [35].

The RNALocate database [7] collected results obtained with all these methods, while lncATLAS [36] focused on subcellular RNA sequencing (subcRNAseq), described below.

RNA-seq and SubRNAseq

RNA sequencing (RNA-seq) is quantitative technique in which the transcripts are first converted into a pool of cDNAs, which will constitute the sequencing library, by RNA fragmentation, adapter ligation, cDNA synthesis, size selection and limited cycles of amplification. In the case of lncRNAs that are generally expressed at low abundance, RNA can be fractioned from different cellular compartments prior to sequencing to increase the relative abundance of unique transcripts. [37]. High-throughput sequencing of the subcellular RNAs can then be used to reveal the identity, abundance, and subcellular distribution of transcripts. Subcellular RNA sequencing (SubRNAseq) yields high-throughput and quantitative data, although the absolute counts of RNA molecules per cell are lost.

1.3. Investigating lncRNAs secondary structure

Several recent discoveries have underlined that RNA structure and function are closely related [38]. The RNA structure is characterized by several modules originating by the interactions among base pairs that can be distant in the linear sequence but proximal when the polynucleotide chain folds upon itself.

Two or more consecutive base pairs forms a stem, which is an intra-molecule double strand. Unpaired nucleotides within a stem form an internal loop. A single-nucleotide asymmetric internal loop is a bulge. An external loop of unpaired bases at the end of stems is an hairpin loop. A junction, or cruciform, is a motif that connects three or more stems. Pseudoknots are intertwined motifs that form when at least two stems are connect by a shared single strand or loop. Hence, RNA secondary structure is complex but can be rationalized in terms of basic modules (or motifs) combined in complex ways. In our work we take into account only simple modules such as internal loop, hairpins, stems and bulges but we strongly consider the size of each motif, that may be a marker of the link between biomolecules (Figure 2).

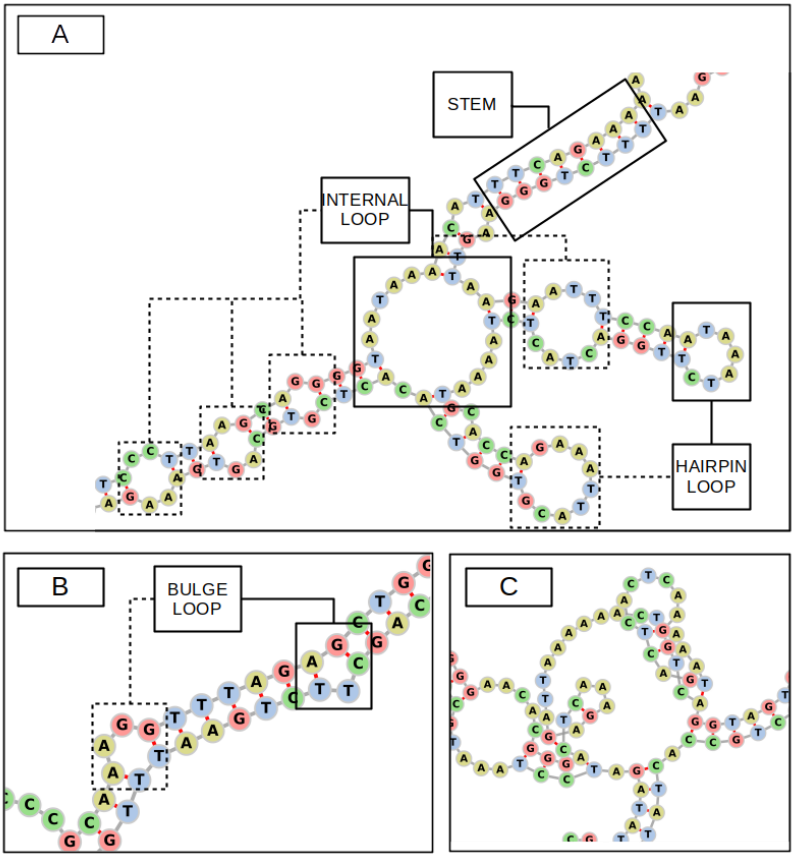


Figure 2: RNA secondary structure motifs. (A) Internal loop, stem, hairpin loop ; (B) bulges; (C) example of complex motif. [Figure obtained with forna]

However, RNA secondary structure is a complex problem, both experimentally as well as in terms of *in silico* prediction. One primary problem associated to the lncRNAs is that they are often large and highly dynamic in living cells, thus their structures are very challenging to solve. The most conventional methods to study RNA tertiary and secondary structure are crystallography, nuclear magnetic resonance (NMR) and cryo-electron microscopy (cryo-EM). Then protocols with *in vitro* and high-throughput applicability have been designed. In 2010, Underwood et al [39] proposed the fragmentation sequencing (Frag-Seq), an enzymatic method based on the cleavage of single stranded RNA by the nuclease P1, followed by an high-throughput sequencing. The demonstration of the power of transcriptome-wide analysis of RNA structure was the development of Parallel Analysis of RNA Structure (PARS). PARS uses both single (nuclease S1) and double-stranded (nuclease V1) nucleases to digest motifs in RNA and to generate a structure score (the preference of each nucleotide in a specific RNA to be single- or double-stranded). Another in vitro

technique is Selective 2'-Hydroxil Acetylation by Primer Extension (SHAPE) [40]. In SHAPE the key idea is that the 2'-hydroxyl group nucleophilicity is different between unpaired and base-paired or otherwise constrained nucleotides, in particular single-stranded or flexible RNA regions exhibit higher reactivity than RNA nucleotides engaged in base pairing or other interactions. Therefore, it is possible using hydroxyl-selective electrophiles such as NMIA that, reacting preferentially with the 2'-hydroxyl group in flexible nucleotides, form a stable 2'-O-ester adduct. NMIA are then inactivated via hydrolysis leaving an unreactive product. By this workflow, SHAPE gives an indication of local nucleotide flexibility.

In vitro experiments, necessary when using nucleases that cannot enter the cell, require RNA purification that must be later renatured to achieve a stable conformation, which might be different from the biologically relevant that the RNA had *in vivo*. Moreover, RNA structure *in vivo* is likely to be more complex, and probably influenced by the binding of small molecules, and interactions with numerous RNA-binding proteins within the cell.

For this reason, in recent years, *in vivo* approaches have been developed. The strategies to study the structure *in vivo* can be classified in two groups:

1. based on chemicals characterized by different reactivity for single-stranded or double-stranded nucleotides. (DMS-Seq and icSHAPE). In these methods, small size chemicals react and covalently modify solvent accessible nucleotides.
2. based on ligation to directly identify the two strands of RNA duplexes. This can be done by a chemical cross-linking as in SPLASH and PARIS or by UV-cross linking such as in CLASH and hiCLIP protocols.

DMS-Seq

Dimethyl sulfate (DMS), introduced for RNA structure mapping in 1980, is one of the oldest chemical reagents used to probe RNA structure. It is base-specific and can alkylate the Watson-Crick face of adenosine and cytosine, as well as the N7 position of guanosine when not base-paired. DMS-Seq combines DMS methylation with NGS.

icSHAPE

The SHAPE technique, previously described, was successfully applied to living cells with the choice of 2-methylnicotinic acid imidazolide (NAI) and 2-methyl-3-furoic acid imidazolide (FAI) as reagents [10]. Transforming NAI into NAI-N3 by the addiction of an

azide to the nicotinic acid ring at position 2, is at the basis of another technique named icSHAPE, in which flexible RNA is acetylated with the SHAPE reagent NAI-N3 and followed by experimental and computational steps. Hence, like DMS, icSHAPE, is a chemical approach that can measure RNA flexibility in cells.

SPLASH – PARIS – LIGR-Seq

To study all RNA duplexes in the cell, methods based on cross-linking the two arms of RNA duplexes using a psoralen derivative were developed. The most commonly used chemical to determine the base pairing relationships is psoralen, a photo-cross-linker that reversibly reacts with staggered pyrimidines on opposite strands. The psoralen-cross-linking-based methods include sequencing of psoralen-cross-linked, ligated, and selected hybrids (SPLASH) [17], psoralen analysis of RNA interactions and structures (PARIS) [41], and ligation of interacting RNA followed by high-throughput sequencing (LIGR-seq). PARIS combines four critical techniques, psoralen cross-linking, 2D gel purification, proximity ligation, and high-throughput sequencing

Despite the innovations in this area, the structural domains that drives the interactions with the other biomolecules are not still well known and difficult to define. In addition, all methods are expensive in terms of cost and manual labor, hence the importance of developing tools that can predict the secondary structure from an RNA sequence. Furthermore, since the secondary structure has a role in the functionality and it is also determined by the primary sequence, it is theoretically possible to establish a direct link between the primary sequence and function, once the secondary structure with its elements has been accurately determined. In our case, the inspiration comes from the Natural Language Processing and the basic idea that the secondary structure from RNA sequences can be represented as a string, and thus can be considered a text.

In this context, several computational frameworks are emerging that demonstrate the potential of the application of automated speech recognition to biomolecules [42]. As described in Material and Methods (Chapter 4), we chose RNAfold to obtain the secondary structure. We then applied the BEAR encoding to describe the secondary structure as a string of characters, while also keeping an informative description, and add a further step in order to reduce the complexity by using the more compact quickBEAR alphabet. The last two steps allowed us to have an optimal input for a text processing approach and a reasonable number of characters combination that lead in turn to a faster algorithm. The aim is to extract

and infer specific features once given two datasets (e.g. positive vs negatives) differing by some biological characteristic. The obtained models can then be used for predicting the class to which a new RNA, not included in the initial dataset, belongs to, and to understand the features leading to the dataset discrimination.

2. Current bioinformatic approaches in studying lncRNAs

2.1. Introduction

In silico methods can be of primary importance for the characterization of lncRNAs and to overcome the standard drawbacks such as the low abundance of data, the cost of labor and the time of experiments that typically characterize them.

They often start from experimental data collected in databases. Since each database usually focuses on one single aspect of the biological problem (i.e. localization, tissue expression or interactions), one strategy is to collect all available data and combine them in order to elucidate lncRNAs functionality. For example, an idea is to add the expression values, subcellular localizations and secondary structures to the interactions map and use them to create an integrated network to give an overall view, as schematically represented in Figure 3.

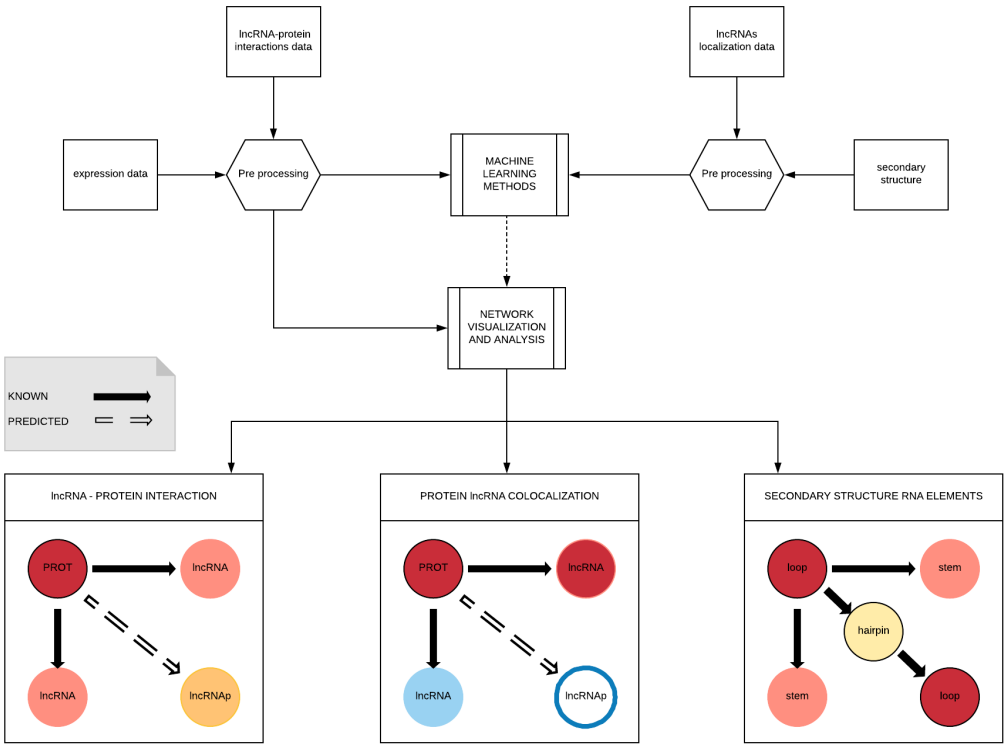


Figure 3: Flowchart of the analysis pipeline

The main problems to face in any pipeline are the input format, the choice of prediction tool and finally the output visualization. This chapter outlines the more widely used approaches in these research areas.

Representation of biomolecules as strings

The protein or RNA primary structure is the sequence of amino acids or nucleotides, respectively, while the secondary structure describes the intra-molecular hydrogen bonds patterns, and finally the tertiary structure represents how these molecules are folded in a 3D space. The description of these biological molecules as strings of characters is facilitated by the facts that they are linear (i.e. there are no ramifications) and have a clear polarity (from the N-terminal to the C-terminal end for proteins, or from the 5' to the 3' end for nucleic acids). The primary sequence is generally based on a 4-letter RNA alphabet (or sometimes a 5-symbol alphabet that includes the unknown nucleotide X [39]) or a 20-aminoacid protein alphabet. More specifically for protein sequences, Suresh et al. [43] introduced a simplified 7 symbol alphabet, whereby the 20 amino acids were clustered into 7 groups based on their dipole moments and side chain volume: {A,G,V}, {I,L,F,P}, {Y,M,T,S}, {H,N,Q,W}, {R,K}, {D,E} and {C}. The idea behind grouping of characters comes from the need to maintain a low dimensional space when describing the primary sequence with the purpose of training a model able to generalize sequence characteristics allowing the usage of the trained model for the inference of labels (e.g. describing some functional feature) associated to biomolecules. This alphabet reduction is crucial in building the prediction model in a reasonable time and in facilitating the identification of patterns, since amino acids in the same group share physical and chemical characteristics. Each sequence is then split in small portions named *k*-grams where *k* is an optimized size, and ideally it represents the fragments (*k*-mers) that have the most influence on the prediction. The best value of *k* is not known *a priori*, and depends on various parameters such as the employed model or the target partner, and each time the optimal value needs to be investigated. Importantly, it has a relevant impact on the running time. For instance, considering 3-mer strings under a 20-symbol alphabet implies exploring 20^3 possible different combinations.

Grouping nucleotides to reduce the alphabet size does not come as naturally, and in principle would have less impact, since the alphabet is already small. Yet, it might be crucial for the description of the secondary structure. The secondary structure of an RNA is the pattern of hydrogen bonding between bases along the polynucleotide chain and depicts the tendency of some nucleotides in the single strand to pair and form complicated structures, from here

on named Secondary Structure Elements (SSE). Some methods has been developed to predict an RNA secondary structure given its sequence, among them RNAfold from Vienna package [44] or RNAstructure [45], which are based on thermodynamic models, predicting the lowest free energy secondary structure as well as, for some algorithms, a number of suboptimal structures. RNAshapes [46] in contrast is based on the abstract shapes approach. To represent in a compact way the RNA secondary structure, the standard notation is the so-called *dot-bracket*, in which unpaired nucleotides are depicted as dots, and paired nucleotides as round brackets. Each nucleotide pair involved in a bond is depicted as an open and a close bracket. Under some assumptions (e.g. ignoring the possibility of pseudo-knot formation), a string of dots and brackets describes unambiguously a distinct secondary structure, and each open bracket can be associated unambiguously to only one close bracket, thus identifying bonding partners. This simple representation, while commonly used by most secondary structure prediction algorithms, lacks information on the structural context; for example, a dot might represent an unpaired nucleotide in an hairpin loop, a bulge, or a nucleotide in an internal loop, and it is not possible to discriminate among these different structural contexts directly without post-processing the dot-bracket string.

As a consequence, a potentially more useful approach would be depicting the secondary structure always as a string, but composed by characters belonging to more complex alphabets, taking into account not only the base-pairing status of each nucleotide but also some more complex structural features.

Recently, Adjeroh et al. [47] presented a Protein-RNA interaction method in which the RNA secondary structure is described as a string-sequence of basic SSEs. They identified three length categories for each element and reported their distribution, suggesting that these can be discriminative parameters for the interactions and functional roles of an RNA.

Heller and coworkers converted the dot-bracket output of RNAshapes [46] by the *forgi* python package [48] and applied this representation to predict an interaction. The string encodes the structural context of each nucleotide in the input sequence with a symbol related to exterior loop, internal loop, stem, hairpin and multi-loop. In this context, Mattei et al.[49] introduced an 85-characters based alphabet named BEAR alphabet (for more details see the Materials and Methods Chapter). It was successfully applied to the Rfam database [50] to compare RNA structures, classify RNAs into families and to discover recurrent structural motifs from a set of unaligned RNAs.

The same principles can be also useful for the description of protein secondary structures. For example, Adjaroh et al. [47] used the Ramachandran code, and more frequently the protein secondary structure is depicted in function of short structural fragments called protein blocks that seem to provide a more accurate representation than classical three state protein secondary structure (helix, sheet, loop).

Once a biomolecule is described in a formal way, taking into account its primary and/or secondary structure with different and appropriate encodings, it becomes easier to apply machine learning to tackle biological problems. These problems must be generally converted in a form of classification (i.e. inferring to which of two or more distinct classes a molecule is more likely to belong to) or regression (extrapolating or interpolating the value of a function given a molecule representation). Again, how a biomolecule is represented is crucial for the performance of the task, in terms of accuracy of prediction and execution times. Describing a biomolecule in a simple but at the same time informative way should therefore lead to models that are more effective. Moreover, the features describing the biomolecule should also be easily available, in order to broaden the applicability spectrum. For example, features derived from the tertiary structure might be very informative, but the tertiary structure is not known, or could be difficult to infer, for most proteins and even more for RNAs. Therefore, a classification or regression model based on tertiary structure features could be accurate but it can be applied only in a limited number of cases. Taking everything into account, informative string representations of primary and secondary structures for proteins and RNAs are particularly attractive, since they maintain the intrinsic simplicity of a string, but they can also include detailed information and have a broad application spectrum.

2.2. Machine learning approaches

There are several ways to apply machine learning to molecules described as strings, most of them based on counting the absolute or relative frequencies of individual characters or of *k*-mers (substrings of consecutive characters of length *k*). In September 2018, Calabrese et al. [51] developed SEEKR, an algorithm based on *k*-mers suited to detect similarities between evolutionarily related lncRNAs. Next, they carried on a network-based approach, demonstrating the possibility to cluster lncRNAs into communities of related *k*-mer profiles. Finally, they examined the lncRNA subcellular localization and protein associations, in order

to investigate whether *k*-mer content correlates with these features. They concluded that *k*-mer content provides information about the subcellular localization of an lncRNA, and for some proteins (e.g. HNRNPC, KHDRBS1, QKI) motif density plays a dominant role in determining RNA binding *in vivo*.

One relatively unexplored way, especially for RNAs, is to consider a molecule as a text. In general terms, the goal of viewing a sequence as text is to identify words (i.e. recurrent substrings of the molecules having a meaning), how words are organized into sentences (i.e. sets of words organized into higher-order patterns), and possibly other features such as syntax, punctuation, and so on. There are currently a number of machine learning methods developed for text processing, but their extension to biological cases is still at the beginning, thus requiring novel approaches. For example, what is a word in an RNA or protein sequence, and what is a sentence, are not trivial to rationalize, and the issue becomes even more daunting when the secondary structure is taken into account. Below, we briefly describe the currently employed ways to face these problems.

The general machine learning task is to learn a target function (f) that best maps input variables (X) to an output variable (Y): $Y = f(X)$.

Features can be of different sorts. They might be continuous (e.g. real or integer-valued such as occurrences of short pieces of a sequence) or categorical (e.g. GO terms, RNA category, localization). A problem with modeling text is that techniques like machine learning algorithms prefer well-defined fixed-length inputs. Machine learning algorithms cannot work with raw text directly; In particular, if the input is a textual data (e.g. a sequence encoded as a text), the text must be at some point converted into numbers, more specifically vectors of numbers, each unit of the vector having a specific meaning.

There are many different ways to overcome these problems and feed a text representation to a machine learning system. The most important are listed below:

- **bag of word representation,** in which the occurrence of each word is used as a feature. It is intuitive and simple but the word order information is discarded, hence the name “bag”, and since the set of potential features is made from all the words that appear at least one time, the dimension of the problem is high. The input of a Bag of word model takes into account only whether known words occur in the document, not where within the document.

- **Phrase-based representation,** in which a feature is generated from contiguous words. The main advantage is that phrases are more informative than single words since there is the additional contribution of the context in which a word is found.
- ***N*gram-based representation,** in which each feature represents a fixed length sequence of size *n* of contiguous typographic symbols. This is applicable to any sequence and it is the only strategy currently applied to biomolecules. It implicitly takes into account semantic or grammatical information but it adds noise, it implies the choice of *n* that it is not intuitive and when *n* grows the dimensionality becomes quickly very high. For example, considering an RNA described by a 4 letter based alphabet, a 5-mer needs a bit vector of dimension 4^5=1024.

Given the representation for the sequences and secondary structure for biomolecules and after encoding them as feature vectors, it is possible to choose the appropriate prediction models.

One of the first methods for predicting non coding RNA complexes using machine learning was reported in 2011 by Pancaldi and Bähler [52]. They trained RF and SVM classifiers using more than 100 features such as GO terms, chromosomal position, physical properties and protein localization. Thereafter, catRAPID [53] was developed by exploiting the physiochemical properties, hydrogen bonding and van der Waals properties as well as the secondary structure. Next, Lu et al. [54] proposed lncPro based on three types of classical protein secondary structures, hydrogen-bond and van der Waals propensities as well as six types of RNA secondary structures.

Muppirala et al. [55] proposed their string-based method RPISeq based on RF and SVM classifiers, while RPI-Pred used only SVM but differs from the previous for the introduction of the 3D protein structure and RNA features.

The methods above require many data that are not always available and, additionally, they have the main drawback that a specific step is required in order to obtain a fixed-length input. As said before, the most popular approach to address this problem is to divide the string in smaller portions named *k*-mers, another is to reduce the space by Fourier series transformation (as implemented in catRAPID and lncPRO).

To date, deep learning approaches are becoming very popular in the bioinformatics area. They usually need a huge training dataset to perform correctly and the core of the structure is a neural network. Examples are DeepBind [56], which was applied to determine sequence

specificities of DNA and RNA binding proteins, and IPMiner [57], which used a quite complex structure based on a stacked autoencoder and a subsequent step in which the extracted features are fed into random forest models.

So far, SVM and RF are the most used models. Importantly, the latter allows the analysis of the features most frequently used by the classifier to predict the outputs (i.e. RPISeq), which might allow a deeper understanding of the patterns detected by the algorithm, and their translation into biological knowledge

However, one of the most attractive research area is Natural Language Processing (NLP), which is an approach used to analyze human languages. It considers the hierarchical structure of the language: several words make a sentence and a sentence transmits a meaning.

In general, it is based on:

- Tokenization: the process of demarcating sections of a string of input characters
- Syntactic analysis: with the aim to analyze a string of symbols conforming to the rules of a formal grammar
- Semantic analysis: the formal analysis of meanings

The current challenge is to describe a biological sequence as a sentence and map it to a vector. So far, the basic philosophy is to apply a variable-length *k*-mer sliding window along the sequence. We tried to exploit different methods, some including only a depiction of the RNA primary sequence as well as others based only on their secondary structure. Finally, we came up with a novel approach that combines both levels, that proved to be more effective and that is described in detail in the Materials and Methods Chapter.

2.3. Data visualization

Another important feature that our approaches can provide is that they facilitate the modeling, building and also description of complex relationships among biological molecules, and within the biomolecules themselves. The inherent variability of biological data, data inaccuracy and noise, the overload of information and the need to study the dynamics and network topology over time, are well-known problems in system biology. One way to overcome these problems is the graph theory. A graph, also called network, is a mathematical representation composed by a set of vertices (V), called nodes, which are

connected by links called edges (E). Formally, the graph is defined as G = (V,E). The exact meaning of the nodes and edges in a graph depends on the specific application and, depending on the application, the edges sometimes have weights, which indicate the strength (or some other attribute) of each connection between the nodes. Moreover, a graph can be undirected or directed. Undirected graphs have edges that do not have a direction and the edges indicate a two-way relationship. On the contrary, directed graphs have edges with direction and the edges indicate a one-way relationship. Mathematically, they are defined as an ordered triple G = (V, E, f), where f is a function that maps each element in E to an ordered pair of vertices in V. Directed graphs are mostly suitable for the representation of schemas describing biological pathways or procedures which show the sequential interaction of elements at one or multiple time points and the flow of information throughout the network.

Within the fields of biology, protein-protein interaction (PPI) networks, biochemical networks, transcriptional regulation networks, signal transduction or metabolic networks are the highlighted network categories in systems biology, often sharing characteristics and properties. The topology of the network often reveals information about its biological significance. In fact, networks follow patterns and rules that allow scientists to go through a deeper investigation towards knowledge extraction. Following the same reasoning, it is also possible to model and describe single biological sequences as networks, considering motifs as vertices linked by weighted edges, whereas the weight is a specific attribute. The objective, in this case, could be the detection of the most frequent motifs within a specific dataset (e.g. lncRNAs in nucleus or cytoplasm).

Many of the available approaches are limited to the analysis of features in a single homogeneous network, considering entities of the same type or domain (e.g., a protein-protein interaction network or a gene network). However a number of works [1] have shown that it is possible to combine different types of interactions and data (e.g. protein-protein interactions, lncRNAs expression similarity, lncRNAs-protein interactions) in order to reveal hidden properties and features and hence investigate their functionality. Yet, it is always needed to calculate a relatedness score for each lncRNA-protein pair in the heterogeneous network. A whole range of different approaches to this problem is available. The most commonly used approaches are guilt-by-association (GBA), the Katz method [58], Combining dATa Across species using Positive-Unlabeled Learning Techniques (CATAPULT) [58], Random Walk with Restart (RWR), and LncRNA-protein Interaction prediction based on Heterogeneous Network model (LPIHN) [59] and HeteSim [60]. The

KATZ measure is a weighted sum of the number of paths in the network that measures the similarity of two nodes. CATAPULT is a supervised machine learning method that uses a biased support vector machine where the features are derived from walks in a heterogeneous gene-trait network. RWR is a method for prioritization of candidate genes by use of a global network distance measure, random walk analysis, for the definition of similarities in protein-protein interaction networks and it add weight to the assumption that phenotypically similar diseases are associated with disturbances of subnetworks within the larger protein interactome that extend beyond the disease proteins themselves. LPIHN is a network-based method that implements a random walk on a heterogeneous network. PRince [61] is a global method based on formulating constraints on the prioritization function that relate to its smoothness over the network and usage of prior information. HeteSim is a path-based measure in which the key idea is that similar objects are more likely to be related to some other objects.

As aforementioned, the graph theory was used also to represent RNA secondary structures [13].Waterman pioneered the graphical representation of RNA in 1978 with the aim of analyzing the secondary structure of tRNAs. Specifically, he depicted the RNA secondary structure as a planar graph and analyzed base pairing in an adjacency matrix. In 1980, Nussinov [62] also developed an ordered label-free representation to compare secondary structures of RNA. In 1990, Shapiro [63] used a tree representation of RNA secondary structure to measure secondary structural similarities. He developed an algorithm for analyzing multiple RNA secondary structures by multiple string alignment. In particular, he defined the tree edit distance between two tree secondary structures to quantify the minimum cost (insertion, deletion, and replacement of nodes) along an edit path for converting one tree into another. This measure is implemented in the RNAdistance program of the Vienna RNA package, widely used to compare two RNA structures. Morosetti [64] further studied similarities in tree graph representations by using topology connectivity indices known as the Randíc index.

In 2003, Schlick and coworkers developed dual graphical representations of RNA secondary motifs in addition to tree graphs in a framework coined RAG (RNA-As-Graphs). In this representation a node is a double-stranded helical stem with more than one base pair; an edge represents a single strand that occurs in segments connecting secondary structural elements such as bulges, loops, and junctions [65].

Thus, the meaning of the nodes or edges used in a network representation depends on the type of data used to build the network and it is important to emphasize that directed or undirected edges can also have a quantitative value associated with them that can convey how reliable the interaction is or how closely related two RNAs are in terms of sequence similarity or the distance of two motifs along the sequence. The data sources can be manual curation of scientific literature, high-throughput dataset or computational predictions that use experimental evidence as their basis and aim to predict unexplored relationships between biological molecules.

3. Materials and Methods

3.1. Introduction

The motivation of this project comes from the fact that despite the growing number of databases, little basic knowledge exists about the normal lncRNAs.

In fact, lncRNAs can interact with DNA, proteins or other RNAs, have tissue-specific peaks in expression, have a variable subcellular localization [20] and it is widely accepted that its functionality is intimately linked to the formation of specific secondary and tertiary elements.

Then, we wondered whether, starting from curated databases, it could be possible to infer the lncRNAs functionality.

To address the problem we referred to the following sources:

- **NPinter v3.0**: It includes interactions between noncoding RNAs and proteins, other RNAs and DNAs, all experimentally verified. The interactions are both physical interactions retrieved from publicly available high-throughput experiment results, and manually collected from publications, and subsequently curated by an annotation process against known databases including NONCODE, miRBase and UniProt. [11]
- **RNA–protein Association and Interaction Networks (RAIN) v1.0**: RAIN is a resource of ncRNA-RNA and ncRNA-Protein interactions that integrates heterogeneous evidence from experiments, predictions, text mining and expert curation [5].
- **Genotype Tissue Expression (GTEx)**: The GTEx data resource consists of whole-genome sequence and RNA sequences and expression estimates from different tissues retrieved from adult donors. [https://gtexportal.org/home/]
- **RNALocate**: It documents subcellular localization in 65 organisms (including *Homo sapiens, Mus musculus* and *Saccharomyces cerevisiae*) for 9 RNA categories. Each subcellular localization entry available on the web page contains detailed information on RNA symbol (i.e. the official name of the RNA), RNA category, aliases, organism, sequence, homology, subcellular localization, tissue, validation method, PubMed ID, detailed description and network [7].

- **lncATLAS**: lncRNA localization in human cells based on RNA-sequencing data sets (subcRNAseq), produced by the ENCODE Consortium. Each entry contains a relative concentration index (RCI), calculated for cytoplasm and nucleus (CN-RCI), defined as the log-ratio, between the two compartments, of the concentration of a given RNA molecule per unit mass of RNA [36]
- **lncSLdb**: lncSLdb collects subcellular information for lncRNAs extracted from literature mining. It stores data from 8 species (Human, Mouse, *Bombyx mori*, *Cryptococcus*, *Microtus transcapicus*, Rat, Bee and Fruit fly)
- **LOCATE**: It houses data describing the membrane organization and subcellular localization of proteins from the FANTOM3 Isoform Protein Sequence set. Membrane organization is predicted by the high-throughput, computational pipeline MemO. The subcellular locations of selected proteins from this set were determined by a high-throughput, immunofluorescence-based assay and by manually reviewing over 1700 peer-reviewed publications [http://locate.imb.uq.edu.au/]

A network approach was applied to have a complete view of the relationships between the biomolecules. The aim is to get a significant and reliable dataset.

Furthermore, it was also of interest to add predicted subcellular lncRNAs localizations, where not available. We investigated whether this aspect could be partly explained by the presence of short motifs in the primary sequence and by a contribution of the secondary structure elements. We therefore developed a new approach to predict the subcellular localization.

Hence, this chapter is divided in three main parts: the first part explains the data sources, the second is focused on the network analysis and the third part describes our approach to investigate biomolecules at the sequences level.

A network approach was applied to find the signatures that characterized the clusters. First, we reproduced the interaction network by selecting the genes and proteins present in the RAIN database. In order to obtain a highly curated list of proteins involved in pathways that are markers of a specific function and not of a casual and aspecific interaction, we looked at the reported molecular function and filtered out all the terms related to splicing and transcriptional level.

The gene expression dataset was retrieved by the table browser of UCSC Human Genome Browser [https://genome.ucsc.edu/cgi-bin/hgTables] by selecting:

- **Genome**: Human
- **Assembly**: Dec2013 GrCh38
- **Group**: Expression
- **Track**: GTEs Genes

The dataset included the classes in Figure 4

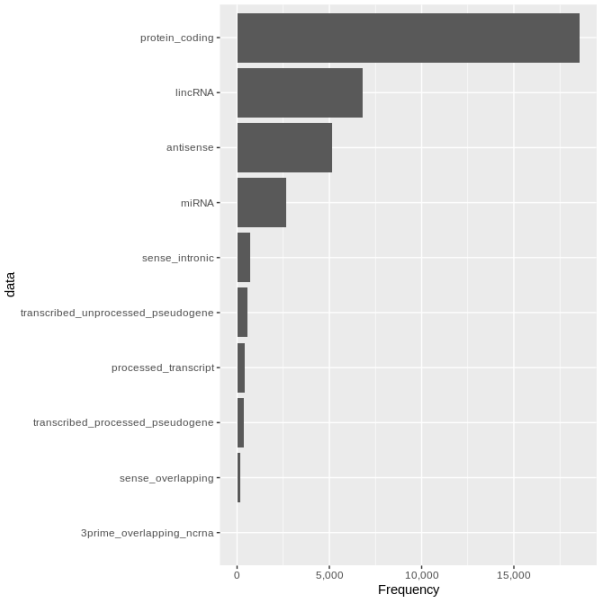


Figure 4: classes included in our analysis about expression

3.2. Dataset of Protein-RNA interactions

The interaction data were retrieved from the download page of the RAIN database [https://rth.dk/resources/rain/download.html]. We considered selected experiments and text mining files. The former collects experimentally supported microRNA-target, ncRNA-protein and ncRNA-ncRNA interactions, the latter includes microRNA-target, ncRNA-protein and ncRNA-ncRNA co-occurrences from text mining (updated weekly).

The file has the following format:

Organism	ID1	ID2	Directed	Evidence	Score	Source	URL	Comment

The organisms are indicated by their NCBI Taxonomy identifier (e.g. *Homo sapiens* has id 9606, *Mus musculus* 10090), while proteins identifiers are equivalent with those in STRING v10 [66]. MiRNAs aliases are equivalent to those in miRBase v20 [29] and aliases of other ncRNAs categories come from Ensembl Biomart v78, thus as ENSEMBL identifiers or, alternatively, the official name is taken as the RAIN identifier.

We retrieved the protein and RNA sequences from ENSEMBL [http://www.ensembl.org/info/data/ftp/index.html/] from the following files:

- Homo_sapiens.GRCh38.pep.all.fa, Mus_musculus.GRCm38.pep.all.fa
- Mus_musculus.GRCm38.ncrna.fa, Homo_sapiens.GRCh38.ncrna.fa.

For the evaluation of our method, we collected two kind of datasets: a randomly generated dataset and a biological dataset derived from multiple experiments collected in RAIN database. For both datasets the RNA secondary structure was predicted by RNAfold. The dot-bracket output was converted into a string of structural context symbols using new_BEARencoder.jar. We used a sliding window to tokenize the sequences and get the words. Since the sequences are of different lengths, we roll the shorter sequence on the longer and alternate the words so created.

3.3. LncRNAs subcellular localization Dataset

We collected data from RNALocate and lncATLAS databases, merging information provided as described hereinafter.

RNALocate

RNALocate provides a file with more than 37,700 entries including 42 RNA subcellular localization, 9 RNA categories (csRNA, lncRNA, mRNA, miRNA, piRNA, snRNA, rRNA, snoRNA and tRNA) and 65 species. The RNA subcellular localization information is manually obtained from articles published and available in the PubMed database before May 2016.

The list of subcellular localizations names was in accordance to the Gene Ontology (GO) Cellular Component (CC) domain.

RNA identifiers were chosen as follows:

- miRbase ids for microRNAs
- NCBI gene and ENSEMBL gene ids for lncRNAs
- NCBI gene and ENSEMBL gene ids for transfer RNAs and snRNAs

Filtering by species (*Homo Sapiens* and *Mus musculus*) and RNA type (lncRNA), we collected the data, organized as described in Table 1

Table 1: Subcellular localization data for each database

Database	Cyto	Nucleus	Cyto/Nucleus	tot
lncATLAS	419	875	0	1294
RNALocate	1041	667	288	1996
lncSLdb	423	599	728	1750
tot	1883	2141	1016	

Then we labeled the lncRNAs as “nuclear” if they belong to the nucleus, “extra nuclear” in the other cases. We obtained three distinct cases:

1. lncRNA in only one tissue
    1.1 same localization
2. lncRNA in multiple tissues
    2.1 same localization in all tissues
    2.2 different localization

In case 1.1 and 2.1 we do not have ambiguities, in the other cases (2.2) we applied a text mining score based on the counts of the number of articles that reported the same localization. We filtered out those lncRNAs for which it was not possible to define a certain localization. We could not apply the same procedure to lncAtlas because of missing information about PMIDs.

lncATLAS

The lncATLAS database collected SubcRNASeq produced by the ENCODE Consortium. RNA-Seq were obtained for a total of 15 cell lines originated from adult and embryological organ sites, including both transformed and normal cells. For each cell, cytoplasmic and nuclear data are available and only for K562 cells subnuclear and subcytoplasmic data are

provided. The localization is defined in terms of the relative concentration of an RNA molecule in the cytoplasm compared to the nucleus, named CN-RCI, where RCI is the log2 transformed ratio of FPKM (fragments per kilobase per million) mapped in two samples, for instance cytoplasm and nucleus. A total of 24,538 genes are included, 17,770 mRNA and 6,768 lncRNAs in at least one cell type, 31 detected in all samples. Among them 150 genes are also present in RNAlocate. lncRNAs of interest are identified by gene names or GENECODE gene identifiers.

Then we labeled the lncRNAs as “nuclear” in case of CN-RCI>0 and “extra-nuclear” whether CN-RCI<0.

lncSLdb

The lncSLdb stores FISH, RNA-FISH and RNA-Seq experiments. While lncATLAS and RNALocate focus on the lncRNAs genes, this source focuses on individual transcripts. The current release [http://bioinformatics.xidian.edu.cn/lncSLdb/download.jsp] contains more than 11,000 entries for 9 species as described in Table 2

Table 2: entries for each species in lncSLdb

Species	Entries
Human	12210
Mouse	2677
Fruitfly	80
Bee	2
Bombyx mori	1
Cryptococcus	1
Transcapiscus	1
Rat	1

The database reports 3 main subcellular locations (Cytoplasm, Nucleus and Nucleus/Cytoplasm), plus some lncRNAs are indicated as accumulated in ribosomes or in chromosomes. Finally, RNAs are identified by their ENSEMBL ID.

3.4. Biomolecules representation

3.4.1. RNA representation

An RNA molecule is a long strand of nucleotide bases. Each nucleotide base can be one of four types (Adenine, Guanine, Cytosine or Uracil), and these are denoted by the letters A, G

C and U. Intra-molecular hydrogen bonds can form between A and U or G and C, and these pairings are generally what is being referred to when a “base pair” is mentioned. The dot-bracket notation is a commonly used method of simply representing the structure of an RNA molecule through open and close brackets, as well as full stops. It is composed by a three-character alphabets that code for an unpaired base ‘.’, an open base pair (BP) ‘(’ and a closed BP ‘)’. However, this simple representation stores no direct information about the structural context of the nucleotide, which must be extracted by means of *ad hoc* post-processing procedures and it is not suitable for machine learning algorithms.

3.4.1.1. BEAR and qBEAR Alphabet

The BEAR is a secondary structure alphabet consisting of 85 characters in which different sets of characters are associated with the different RNA basic structures (loop, internal loop, stem and bulge). Since it contains many non-alphanumeric characters, a grouped alphabet, named quick BEAR (qBEAR) is also used. For each RNA molecule, the sequence is given, followed by the structure, expressed in bracket notation, BEAR and qBEAR alphabet. The correspondence between the two alphabets is described in Table 3.

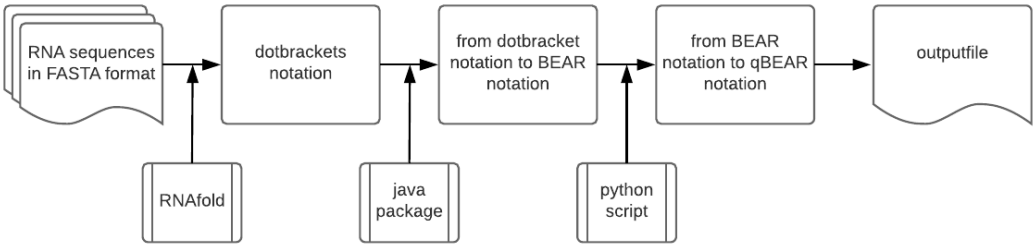
Table 3 Bear-qBear conversion

Bear notation	**qBear notation**
abcde	Z – short stem
fghi	A - medium stem
=	Q – long stem
jklmnopqr	X – short loop
stuvwxyz	S – medium loop
^	W – long loop
!”#$%23456	C – short internal loop
&’()7890	D – medium internal loop
+>	E – long internal loop
[]	B - bulge loop
{}	G - bulge branch
:	T - branch
ABCDE	V – short stem branch
FGHI	F – medium stem branch
J	R – long stem branch
KLMNYZ~?	N – short internal loop branch
OPQRS_/ \	H – medium internal loop branch
TUWYZ@	Y – long internal loop branch

The main advantage of the BEAR alphabet is that it unambiguously associates to each nucleotide in an RNA sequence its secondary structure. Differently from the dot-bracket notation, the BEAR encoding allows to easily discern from the unpaired nucleotides

belonging to a loop and a bulge, for example. The length of the sequence is preserved and it makes directly available the length of the SSEs.

In order to obtain the secondary structure description for each RNA we applied the following steps:



The input is a file in FASTA format, in which

- The line containing the name and/or the description of the sequence starts with a ">"
- The words following the ">" are interpreted as the RNA id
- The second line reports the RNA nucleotide sequence

Then we folded all sequences by RNAfold, included in the Vienna Package. Once obtained the dot-bracket notation, we used the encoder java package BearEncoder.new.jar [http://beam.uniroma2.it/tools/BearEncoder.new.jar] to convert the dot-bracket secondary structure, output of RNAfold, into the BEAR encoding.

The output of BearEncoder is a FastB file with primary sequences, dot-brackets and BEAR-encoded structures. A python script then converts the sequences from BEAR to qBEAR notation. The output is a tab separated file with primary sequence, dot-brackets, BEAR and qBEAR encoded structure. This file is then used as the input to machine learning models.

To tokenize each RNA, we then applied 2 different procedures, here named *kmers-procedure* and *text-procedure*.

In the *kmers-procedure* we computed *k*-mer frequencies, with *k* ranging from 1 to 7, while in the *text-procedure* we designed a different method in order to include the motif length and its context in our prediction.

The basic idea is to convert the RNA sequence into a text, formed by words. With this aim in mind we used several strategies (see Figure 5Figure 5):

1.ps and 2.ps depict the sliding window (*sw-n*) approach. It allows exploring punctual changes along the sequences but creates a huge number of words

2.ss and 3.ss describe two approaches that involve the RNA secondary structure encoded as qBEAR alphabet. The 2.ss approach, here named *splitbychar,* tokenizes the sequence by a single character (e.g. T). The 3.ss method *splitbypattern* tokenizes the primary sequences using as spacer the secondary structure element (the pattern may be the secondary structure encoded to qBEAR notation as showed, as well as the secondary structure encoded to BEAR notation).

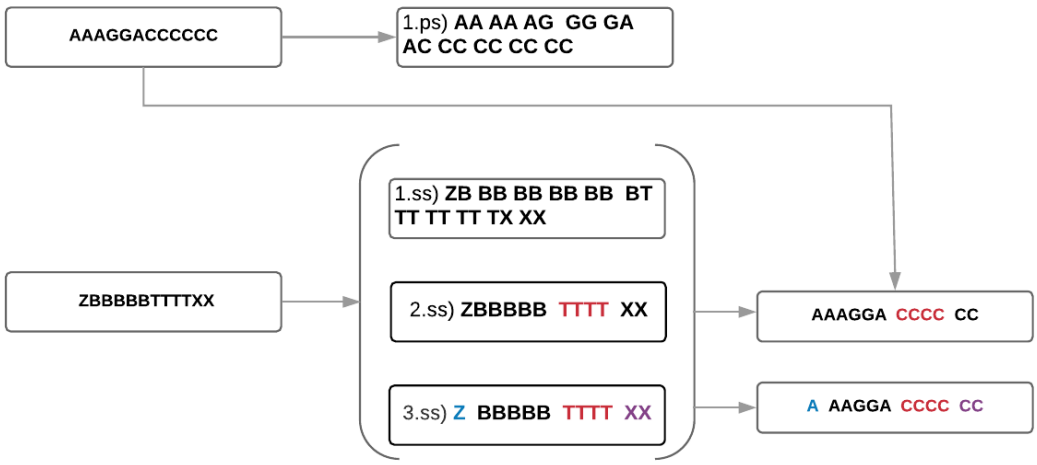


Figure 5: Z is a stem element, B is a 5 nt bulge loop, T is a 5 nt branch, X is a 2 nt length short loop.

We did several tests using all the possible combinations, before choosing the *splitbypattern* strategy (see Appendix A).

3.5. Machine learning models .

3.5.1. Kmers model

We tested a number of popular learning models:

- Random Forest (RF)
- Decision Tree Classifier (DCT)
- K-Neighbors Classifier (KNN)
- Extra Trees Classifier (ExtraTC)
- Ada Boost Classifier (ABoostTC)
- Gradient Boosting Classifier (GBoostC)

Then, each model was optimized running a custom script. The script could be run in three modes:

- The optimization mode: finds the best parameters for each model
- The training mode: It trains the models one established the best parameters
- The Voting model: It combines all models

Table 4 shows the training parameters. In case of KNN we used n_neighbours = 2 while in case of ABoostTC we set also a learning rate 0.1 and as algorithm option “SAMME”.

Table 4 *k*-mer best parameters

	N Estimators	Max features	Min samples split	bootstrap	Max depth	criterion
RF	500	None	3	-.	None	entropy
DCT	-	-	-	-.	None	entropy
EXTRATC	500	sqrt	2	True	None	-
ABoostTC	500	-	-	-	-	-
GBoostC	500	-	-	-	-	-

3.6. Text model

As mentioned, training a model means taking a labelled training example and adjust the parameters slightly in order to predict the training sample label more accurately. How good a prediction model does, in terms of being able to predict the expected outcome, is measured by a loss function.

FastText has three loss functions:

1. The negative sample functions (ns)
2. The softmax function (softmax)
3. hierarchical softmax (hs)

We chose the hierarchical classifier because it reduces the complexity of the model training and testing from linear to logarithmic with respect to the number of classes. In the hierarchical classifier, the different categories or labels are organized in a binary tree. Each leaf node represents a label and every node in the binary tree is representative of a probability. Since each word has a unique path from the root down its corresponding leaf, the probability of picking the word w_i is equivalent to the probability of taking this path from the root down through the tree branches. The advantage is that, instead of computing the probability for each possible label, only the probability of each node on the path to the correct label is computed. Following this idea, the probabilities of each node are the parameters being optimized. The Huffman algorithm is used to build the tree. Every word is depicted as a code. The basic principle is that short words have long codes and long words are represented by short codes. After building the model, new bits of text can be included in the model. The algorithm calculates the probability for every single label and the output is the label associated to the highest probability.

We optimized the algorithm parameters with a Python script. All values are listed in Table 5.

Table 5: FastText model parameters

parameter	meaning	FastText Short form	value
epoch	Number of times each example is seen	epoch	700
learning rate	How much the model changes after processing each example	lr	0.7
dimension	Size of the vectors	dim	70
subwords	subwords contained in a word	minn	3
		maxn	6
N grams	Concatenation of n consecutive tokens	wordNgrams	4
loss function	tells how good our current classifier is	loss	hs

A learning rate equal to 0 means that the model doesn't change at all and thus does not learn anything.

Among the most important parameters of the model there is its dimension, i.e. the size range of the vectors. The larger they are, the more information they can capture, but the training is slower. Moreover, it drastically affects the size of the output, that is the *dimension word vector x words of vocabulary.*

fastText is an algorithm that can examine the context and also learn vectors for subparts of words, which is particularly interesting for building vectors for unknown words. The higher the learning rate is, the faster the model converges to a solution but at the risk of overfitting to the dataset.

The output is the probability

$$P(L|Hs) = \frac{e^{h*VL}}{\sum e^{h*vk}}$$

Whereas,

- VL is the classifier label
- Hs is the feature sequence defined as $Hs = \sum Xw$

Besides text classification, fastText can also be used to learn vector representations of words. In fact it is possible to print word vector representations. In the output *txt* file, each line contains one word represented as a vector in *n* dimensional space, whereas *n* is given by the *dim* parameter (see Table 5). The words in the file are sorted by decreasing frequency (i.e. the first *n* lines are the most frequent words). The size is equal to the *dimension word vector x words of vocabulary.* Indeed, fastText word vectors are built from vectors of substrings of characters contained in it. This allows building vectors even for words that did not appear in the original data by the sum of known substrings.

3.7. Technical Requirements

All analyses in this study were performed by using Python (version 2.7.12), R (version 3.5.0) and the FastText library plus scripts developed *ad-hoc* and the ViennaRNA Package [44].

3.7.1. R Packages

We used Hmisc [67] to calculate correlation matrices, DataExplorer [68] for some data wrangling and plotting, Networkx, threejs and htmlwidget for network analysis and visualization.GO.db [69], org.Hs.eg.db [70], biomaRt [71] were used to annotate lncRNAs.

3.7.2. Python packages

We used pandas toolkit [72] to handle data and scikit-learn package version 0.20 [73] to train the machine learning models

3.7.3. FastText

FastText is an open source tool for text classification. It is usually applied to canonical texts (i.e. in human languages). After having transformed them into continuous vectors, it can be used for any language-related task, and as such it is dedicated to representing and classifying text in a scalable environment. It has been designed to work on a variety of languages, including English, German, Spanish, French, and Czech, but so far never applied to biological alphabets. The library is written in C++ but also has interfaces for other languages like Python [74] as well as Node.js. FastText combines the natural language processing and

machine learning, representing sentences with bag of words and bag of *n*-grams, as well as using sub-word information, and sharing information across classes through a hidden representation.

3.7.4. Performance evaluation

Prediction performance was evaluated by several commonly used metrics, like Precision and Recall:

$$Precision = \frac{TP}{TP+FP} \; ; \; Recall = \frac{TP}{TP+FN}$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, and FN is the number of false negatives. The Precision is the percent of the correctly predicted labels. The denominator is the total predicted positives and thus it reports how many of the predicted positives are actual positives. In fact, the recall (also known as sensitivity) is the percent of labels that is actually recalled over the total labels that actually existed and thus is the percent of labels successfully predicted. In addition, the confusion matrix (i.e. a 2 x 2 matrix reporting TP, FP, TN, and FN) was printed at each step. It provides an indication of the errors made. We also computed the sensitivity versus 1-specificity, which has a better statistical foundation than the other performance measures and that can be used to compute the Receiver Operating Characteristic (ROC) curve. Hence, we finally reported the AUC (Area Under the Curve). AUC values range from 0 to 1: the AUC equal to 1 stands for a perfect classifier instead AUC=0.5 stands for a random classifier.

4. Results

Summary

In silico inference can be suitable to reach the aim of understanding the functional roles of non-coding RNAs, overcoming the typical experimental drawbacks intrinsic to lncRNAs studies, for which data are less abundant and technically challenging. On the other hand, there are other inherent difficulties in bioinformatics procedures. Usually starting from experimental data, *in silico* methods collect information from public databases that do not give the same type of information, and the investigation processes are typically hampered by the lack of comprehensive annotations. In this context, it may be essential a technique able to homogenize the IDs, based on the same and reliable categories, and finally that predicts the missing information. Briefly, our pipeline involves the collection of data depicting interactions and subcellular localizations, primary sequences and secondary structures from several databases, and the creation of a unified resource avoiding duplicates and with a precise nomenclature. In particular, our goal is to include and integrate the following features:

- lncRNAs and proteins expression profiles similarity
- lncRNAs secondary structures
- lncRNAs sucellular localization
- well-known protein-lncRNAs interactions
- well-known protein-protein interactions

We aimed at reconstructing an heterogeneous network with known lncRNAs-protein interactions. The network is heterogeneous because the nodes are different types of molecules. Then, we superimposed the expression correlation network between mRNAs and lncRNAs in which the nodes are proteins and lncRNAs, and the edges are weighted by the expression profiles Spearman correlation value (previously calculated between all possible couples mRNAs-lncRNAs and filtered above a certain threshold, as described in Materials and Methods). In addition, we implemented a machine learning method to infer lncRNA subcellular localization, and another to infer novel protein-lncRNA interactions. We hypothesized that the most central protein nodes in the network should play a central role in combination with the lncRNAs partners.

We started from the gathering and organization of protein-lncRNA interactions from the major specialized databases, in order to assess whether the available data are sufficient to reconstruct an interaction network representative of the processes involving these kind of binding, which could allow the analysis of the network topology and features, and the identification of regulatory modules. We assessed that the available data, mostly extracted from CLIP experiments, are remarkably unbalanced, describing interactions for a limited number of different proteins, and also difficult to interpret and validate, since the reliability of the provided information is not always clear, and since this information is oftentimes inhomogeneous in terms of annotations and identifiers. Hence, we felt that, in order to assess how much a given interaction is likely to be true, and to extend the network with unreported but likely novel interactions, we could use machine learning procedures to filter and infer interactions. This idea raised challenging issues, related on how to represent properly lncRNAs, taking into account their primary and secondary structures, in ways that preserve the available information, but sufficiently simple and systematic that would allow these representations to be fed into a learning algorithm. We explored text-processing methods, which were employed in the recent past for protein description, but not yet extensively for RNAs. We employed a novel way for RNA secondary structure encoding, and for describing a RNA molecule and its structure as a text composed by words. As a test case, we tackled the lncRNA sub-cellular localization, which is important for understanding their biological functions and that can be used as a filter for assess protein-lncRNA interactions. Finally, we applied the same procedures for the inference of protein-lncRNA interactions. In conclusion, the methods described here are simple and accurate enough to warrant their general applicability, to any kind of learning problems related to RNAs.

4.1. The protein-lncRNA interaction network

Network biology allows the representation of biological entities not only as individual components but as an interacting system. Remarkably, the different type of analysis depend on the nature of the information enclosed in the edges (the links between nodes). We started to visualize the interactions retrieved from RAIN database [5] filtered by species and categories, as described in the Materials and Methods chapter. RAIN contains interactions between non-coding RNAs and proteins, and also between RNAs. Interactions are extracted from curated examples, experimental data (mostly CLIP), predictions (for interactions between microRNAs and proteins or non-coding RNAs) and automatic literature mining. To

each interaction, a confidence score is assigned. RAIN includes curated knowledge that comprises well-established interactions from the scientific literature or listed in expert-curated databases. Interactions are collected for nine classes of ncRNAs; experimental interactions are retrieved from miRTarbase [75], NPInter [11], StarBase [76], while predicted interactions are retrieved running miRanda [77], PITA [78], TargetScan [79], miRDB [80] and StarMirDB [81]. The experimental data encompass CLIP, CLASH and CRAC methods (see the Introduction Chapter, paragraph 1.1 for more details). Following the authors guidelines we selected interactions with a confidence score>0.15.

The number of interactions for the four species considered in RAIN is reported in Table 6:

Table 6: number of miRNA–mRNA, ncRNA–protein and ncRNA–ncRNA interactions per organism in RAIN

Species	TAX ID	Score<0.15	Score>0.15
Saccharomyces cerevisiae	4932	99	717
Homo sapiens	9606	661612	190737
Mus musculus	10090	302915	77455
Rattus norvegicus	10116	69742	20393

Using the confidence score as filter, we have in total 22,326 protein-miRNAs interactions and 486,683 protein-lncRNAs interactions. Despite the apparent large number of interactions, these can be mapped to a relatively small number of different proteins, 18,605 in total. From these data, we built an undirected graph in which the nodes are the proteins and the lncRNAs and the edges the physical interactions. In this case, the link between nodes only tell us that A (Protein) binds B (lncRNA). The first drawback in this kind of network is that one protein can interact with many lncRNAs (the opposite, i.e. one lncRNA that interacts with more than one protein, occurs less frequently), leading to a confused view characterized by large hubs. In addition, within the hubs it was difficult to detect false positives basing only on a single feature. In fact, in the resulting network, the distribution of the degree of the protein nodes (i.e. the number of connections it has to other nodes) shows a remarkable imbalance, with a limited number of proteins responsible for the majority of the interactions (Figure 6).

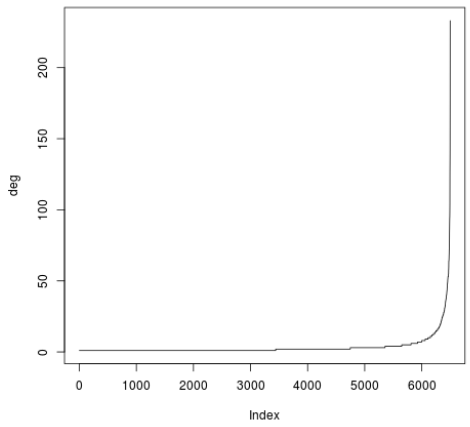


Figure 6: Degree distribution

As a result, the network do not show significant hubs or other topological features usually associated to biological networks (Figure 7).

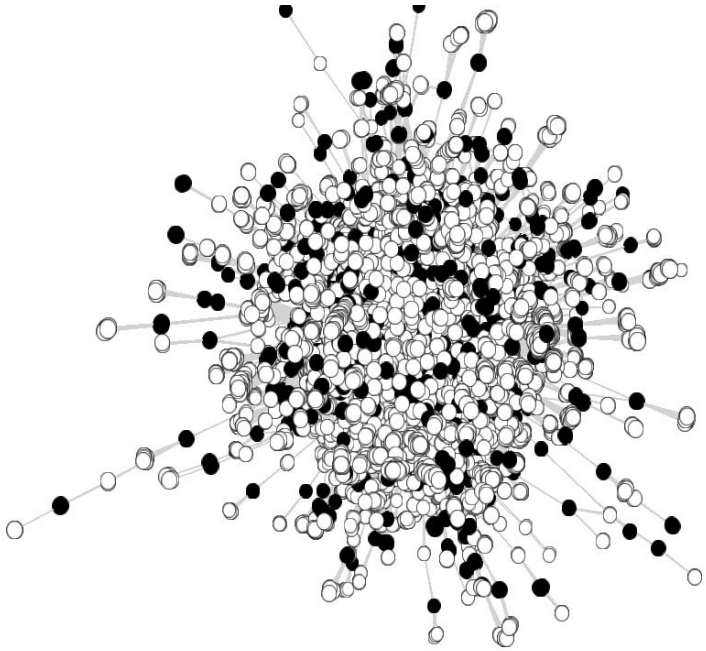


Figure 7: random graph Protein-lncRNAs

Then, we decided to add qualitative and quantitative values in order to increase the reliability of the collected interactions, preferring a systematic approach. As a way to better

characterize the network and, possibly, to filter unreliable edges, we included and integrated two additional descriptors, namely: i) sub-cellular localization; ii) expression profiles. The rationale is that interacting partners must be, at least transitorily, in the same compartment, therefore edges between nodes that do not satisfy this criterion could in principle be discarded. Second, as often occurs for interacting proteins, the expression of a non-coding RNA and its protein partner could be coordinated, and the expression level profiles could be a possible marker for the interactions prediction.

We looked for the protein subcellular localization in QuickGO by R Bioconductor package and we used the RNALocate and lncSLdb databases for RNA subcellular localization. Finally, we merged all the data. At the end of the analysis, we have complete subcellular localization annotation for 3,996 protein-RNA interactions (Table 7).

Table 7: number of entries annotated as cytoplasmic,nuclear or with both localization.

Protein localization	**lncRNAs localization**		
	cytoplasm	nucleus	nucleus/cytoplasm
cytoplasm	584	637	18
nucleus	988	1284	48

The expression values were retrieved by the UCSC portal (See Materials and Methods for more details). In total, the file included expression levels for 52,896 genes that we filtered by the following categories:

- protein coding
- transcribed unprocessed pseudogene
- lincRNA
- antisense
- miRNA
- sense intronic
- transcribed processes pseudogene
- sense overlapping
- 3’overlapping ncRNA

Considering only these classes, we retrieved expression profiles for 32,182 genes, classified as described in Figure 8.

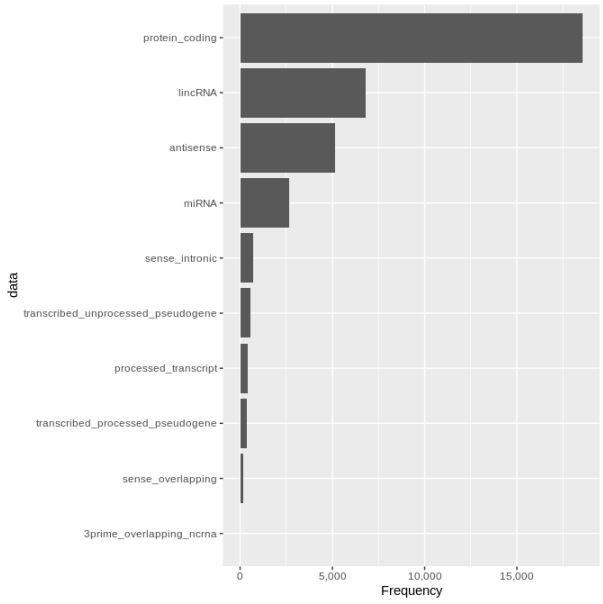


Figure 8: categories after filtering

We referred to the transcription of protein coding genes as proxy for protein expression.
In order to choose the correct correlation analysis, we run a Shapiro Test to test the normality of our distributions. The null-hypothesis of this test is that the population is normally distributed. According to the p-value of the distributions, we rejected the null hypothesis (see Figure 9)

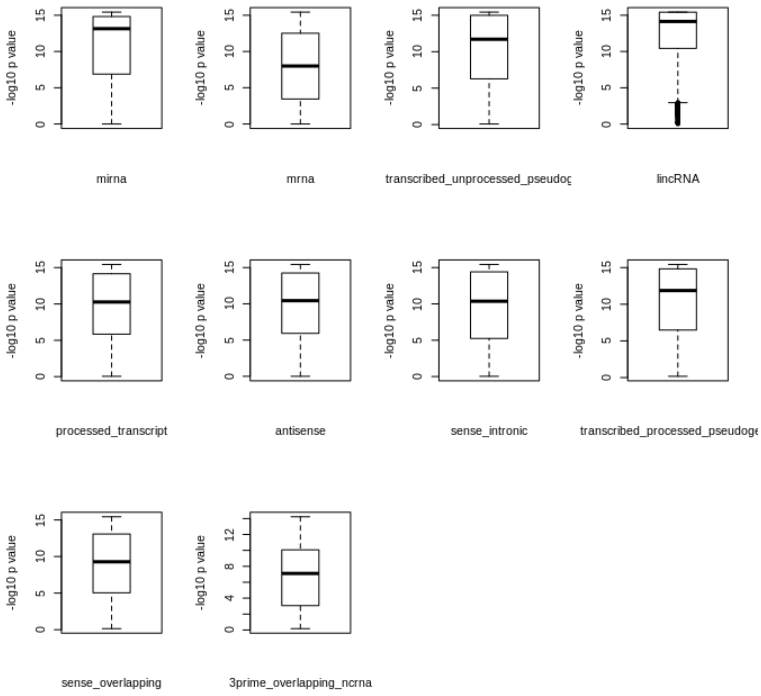


Figure 9: pvalue distribution, Shapiro Test

This means that the widely used Pearson correlation is not suitable, and we decided to calculate the Spearman correlation for each mRNA-ncRNA pairs. Figure 10 shows the Spearman correlation expression density plot for each gene class.

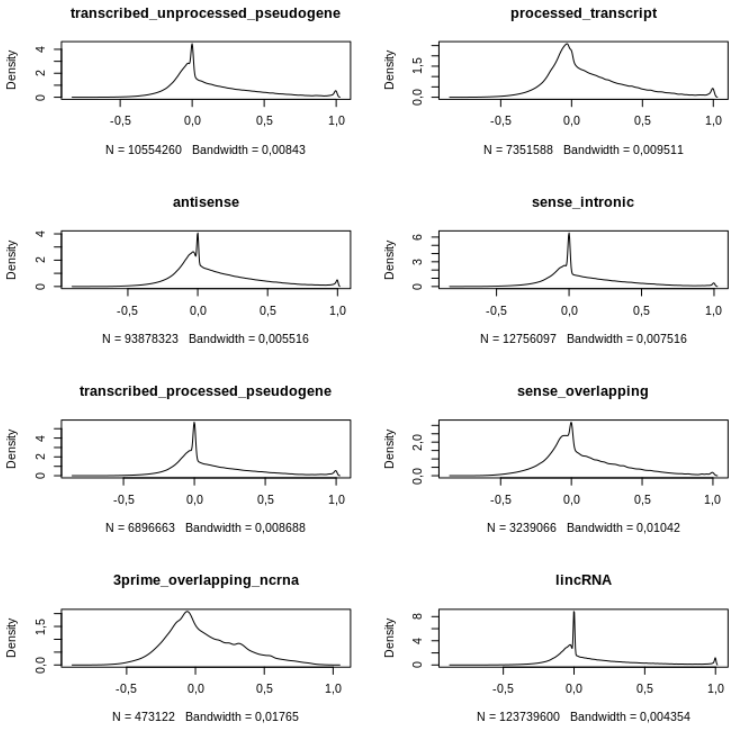


Figure 10: Expression values density plot for each category

We superimposed the expression values over our interaction datasets and collected 761 entries. Among them 262 has a correlation greater than 0.4, and 23 less than -0.4, 476 interacting pairs have correlation values between -0.4 and 0.4 (Figure 11)

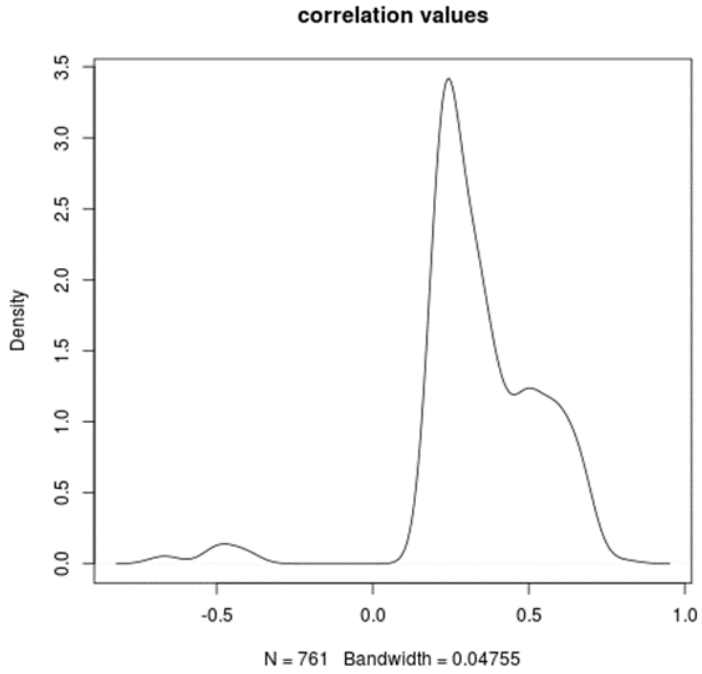


Figure 11: Expression profile Spearman correlation density plot

These results suggest that expression profile correlation between genes for interacting proteins and ncRNAs, while being high in a relatively large number of cases, cannot be easily adopted for filtering and/or predicting interactions.

Next, we superimposed expression levels and known subcellular localization (Figure 12). For only 145 interactions we have complete information about expression and localization. Among them 75 biomolecules are colocalized and positively correlated (Spearman correlation>0.4), while 5 are colocalized and their expression negatively correlated.

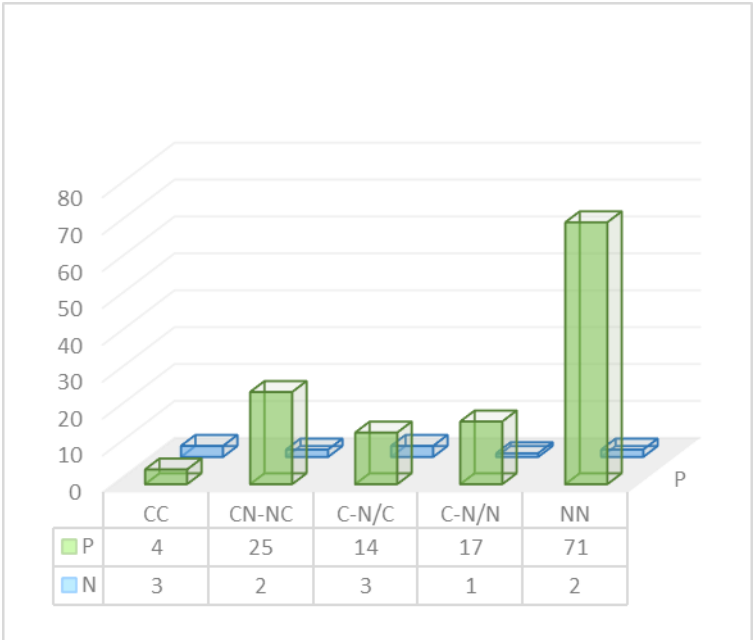


Figure 12: CC : both in cytoplasm, CN-NC: do not colocalize, C-N/C: cytoplasm-Nucleus/Cytoplasm, N-N/C: Nucleus-Nucleus/Cytoplasm, NN:both in nucleus

Hence, in slightly more than half cases the interacting partners are known to have the same subcellular localization. This can be due to the fact that localization data is quite rare and incomplete, especially for RNAs. Therefore, even if this criterion could be useful in principle for filtering the network (a network visualization with all features included is shown in Figure 13), data are not sufficient for doing so in a systematic way. As shown in the next paragraph, we then tried to infer the subcellular localization for lncRNA using learning methods. Subcellular localization inference for proteins was successfully attempted in the past, but working with RNAs raises additional challenges that we had to overcome.

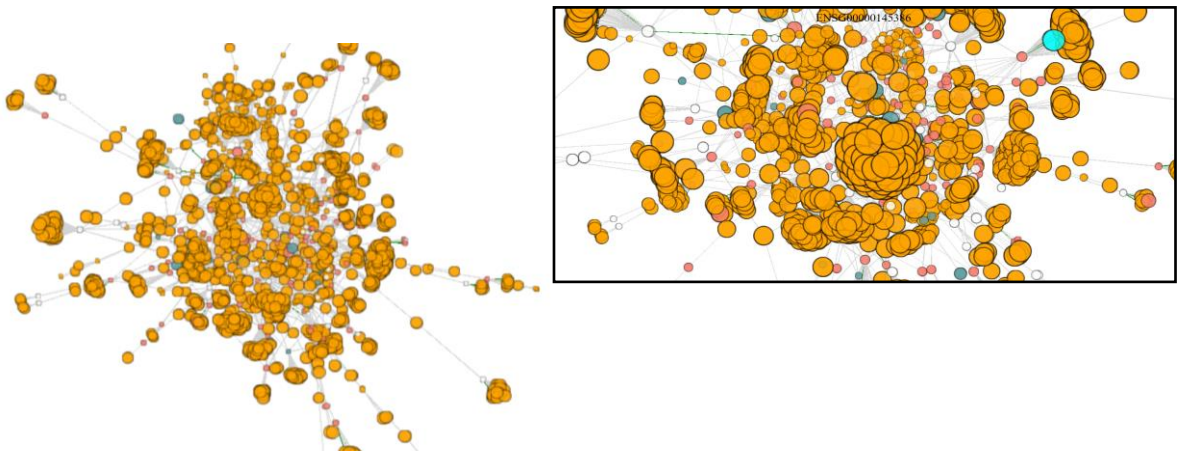


Figure 13: protein RNA interaction as graph in which big nodes are the proteins and small nodes lncRNAs, the dark cyan highlight the cytoplasmic localization, the salmon color is for nuclear localization. White for double localization. On the right a detailed view in which it is shown as elected node in cyan with the ID that appears interactively.

4.2. Prediction of lncRNAs subcellular localization

The key idea of this section is that the lncRNAs function is intimately related to their location in the cell, and that their location in the cell depends from some signals in their sequences and/or structures. Moreover, since there is the evidence that lncRNAs bind to several molecules at each level, and since the binding process requires the presence of the partners in the same cell compartment as well as a certain expression level, investigating the relationship between location, expression and interaction can be of primary importance for the characterization of this category of RNAs. Simply put, the rationale is that the interaction between two biomolecules is more likely if the interaction partners are located in the same environment. Since the interaction between molecules is a complex problem involving multiple causes and effects and possibly related to the primary sequences and secondary structures, we decided to develop a sequence-structure based method.

For this task, we trained learning models to predict lncRNA subcellular localization using for training, testing and validation of the models all data retrieved from the databases lncATLAS, RNALocate and lncSLdb. These datasets include a total of 1883 cytoplasmic lncRNAs, 2141 nuclear lncRNAs and 1016 that are found in both locations (as described in Table 8 and Figure 14).

Table 8 : Number of entries for each database

Database	Cyto	Nucleus	Cyto/Nucleus	tot
lncATLAS	419	875	0	1294
RNALocate	1041	667	288	1996
lncSLdb	423	599	728	1750
tot	1883	2141	1016	

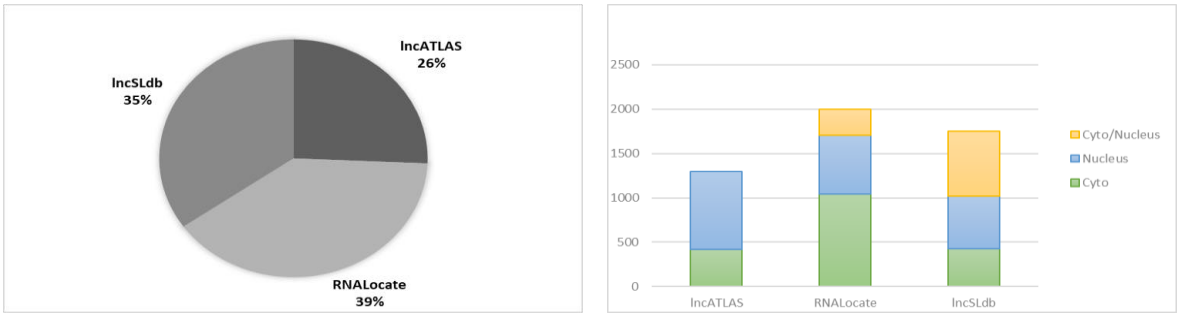


Figure 14: Representation of input dataset. We first considered only lncATLAS and RNALocate

In the first part of our work we focused on lncATLAS and RNALocate databases, since the lncSLdb has been published very recently (September, 2018), and we became aware of its availability only while writing this thesis. When building the dataset, we faced a first issue, because lncRNA genes are often alternatively spliced, so one might wonder whether all splicing variants encoded by a gene have the same subcellular localization. Current databases report the localization data at gene level, and not for individual transcripts. When the expression levels for individual splicing isoforms were available, we kept only the most expressed transcript, otherwise we kept all transcripts, since a gene may encode for many isoform with many different subcellular localization types.

The resulting expression level distribution is a bimodal distribution (Figure 15). Then, we filtered out the transcript with length>10,000 for a few main reasons: First, we saw that the most of our entries has a length<5000 nt but at the same time we lost a lot of molecules that in turn could affect our training; second, we wanted to speed up the folding process that is the slowest step in all pipeline; third, since we used RNAfold [44], we had to face with a limit in length.

Figure 15: on the left: transcript length box plot. On the right: Unimodal distribution before filtering by expression level. We kept only the most expressed transcript obtaining a bimodal distribution

In order to avoid confusing data, we decided to consider only nuclear and cytoplasmic transcripts, not considering the ones found in double locations and the few that localize in other cellular compartments.

Once the dataset was created, the following and crucial step is to encode each molecule, in a way that must be informative and suitable for the learning algorithms. As stated before, we wanted to explore text encoding strategies, but with the additional challenge of including also secondary structure information. To employ text-processing strategies, each molecule must be tokenized, i.e. divided into words. This issue is not trivial for RNA molecules. The primary sequence can be easily divided into sub-sequences of fixed length *k*, using a sliding window that generates overlapping *k*-mers. For secondary structures, how to do this is not obvious. The usual way to represent secondary structures is by the dot-bracket notation, described in the Introduction section, which employs a too limited alphabet (only three characters) that would generate non-informative *k*-mers. We then converted each RNA secondary structure into the BEAR notation, which assigns to each nucleotide a different symbol, based on the type and size of the secondary structure element it belongs to. The full BEAR alphabet is composed by a large number of characters, therefore, when dividing a BEAR string into short *k*-mers there is the risk that very similar structures would be represented by completely different sets of characters, impairing the learning. We used instead a reduced version of the BEAR encoding, dubbed qBEAR (quick BEAR), that should alleviate this potential issue. Therefore, each ncRNA in the dataset was folded using

RNAfold, then encoded with the qBEAR notation, then tokenized using a sliding window (from bi- to epta-mers) and labeled with its subcellular localization (two classes: nuclear and cytoplasmic), and finally we trained a number of popular models, as described in the Materials and Methods section. For each model, we applied a ten-fold cross validation and evaluated the model by the AUC (Area Under the Curve) score. The results are listed in Table 9.

Table 9 performance for proposed methods using the *k*-mer procedure with *k*=2,3,4,5,6,7

Learning models	**kmers AUC values**					
	2	**3**	**4**	**5**	**6**	**7**
KNN	0.6	0.61	0.62	0.61	0.63	0.61
DTC	0.54	0.54	0.52	0.58	0.57	0.61
RF	0.67	0.67	0.68	0.69	0.69	0.77
EXTRATC	0.67	0.68	0.69	0.7	0.69	0.77
ABoosTC	0.57	0.58	0.56	0.56	0.57	0.57
GBoosTC	0.61	0.61	0.61	0.61	0.62	0.71

The AUC remains somewhat similar for the different values of *k*, but there is a general improvement as *k* increases, leading to the highest accuracy when using eptamer tokens, with the only exception of the AdaBoost classifier. The most accurate models, Random Forest (RF) and Extra Tree classifier (EXTRA), both lead to an AUC of 0.77 and are both based on ensembles of decision trees.

The tokenization strategy employed in this first experiment is therefore effective, but a possible limit is that each token is treated independently of its context, i.e. which other tokens could be recurrently associated with its own. When considering a biomolecule as a text, the tokenization produces a set of words, but words in a text are combined into sentences, following specific syntax rules, and the commonly employed learning models are not able to capture this aspect. Moreover, most models are black boxes, which means that it is not easy, and often impossible, to extract from a model what the model has learned. In our case,

from the trained models is not trivial to understand which could be the signals the dictate the localization of a RNA. Additionally, another limiting issue shared by all these models is that they required as input a set of fixed-length vectors. This means that, regardless of the length of the RNA molecule, the same number of features must represent each one of them, while the tokenization procedure will obviously produce a different number of tokens for each molecule. The common solution is to represent each data point with a vector reporting all possible tokens that can be generated for the chosen value of *k* and for the four nucleotides. For example, is *k* is 2, each vector will be composed by units corresponding to all possible dinucleotides (e.g. AA, AC, AG, AU, CA, CC, CG, CU and so on), and the vector units will be set to the absolute or relative frequency of that token in the RNA molecule. This strategy has been proven to be effective, but is less suitable when the considered molecules have a large range of sizes. A small RNA will be encoded by a vector in which a possibly large number of units will have to be set to a value of 0, and these sparse vectors might hinder the training and, as a result, the model accuracy. Finally, the token size *k* must be chosen at the beginning and it is fixed for all molecules and for the whole molecule length, raising two other potential issues: i) it is not possible to know beforehand which is the optimal *k* value, and one must proceed with trial and error; ii) it is possible that different parts of an RNA molecule are better described with different token sizes, therefore being limited to a fixed *k* might prevent a good RNA representation.

We then tested a different Text model, implemented in the fastText software, that is able to capture relations among words. FastText was employed on the same data used for the previous tests, by building the input dataset as described in Material and Methods chapter, splitting in training, test and validation datasets (proportion 60% of the data, 20% and 20%, respectively). The algorithm provides several potential advantages: the context of each token is taken into account, and each data point (i.e. each RNA molecule) is effectively treated as a text composed by words organized into sentences; ii) there is no requirement of having fixed-length vectors to describe each molecule; iii) there is no requirement of having each token of the same size; iv) the most relevant words and sentences, those that have a major weight in the training phase, could be retrieved and used to rationalize what the algorithm has learned. Since we were not limited anymore to fixed length tokens and token vectors, we devised novel ways to tokenize each RNA molecule. The most effective consists in the tokenization of the RNA primary sequence based on the succession of its secondary structure elements. In practice, after the RNA was folded and its structure converted into the qBEAR

notation, we indexed and split the primary sequence each time there is a change of character

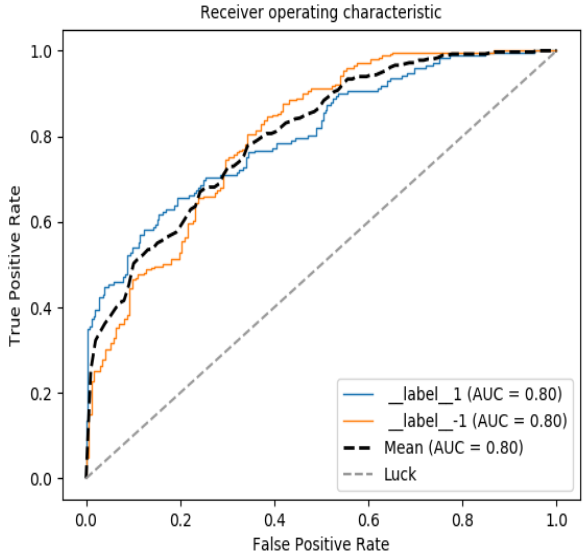


Figure 16: AUC Curve: label -1 for cytoplasmic transcripts, label 1 for nuclear transcripts, better than those obtained with any other learning model

in the qBEAR string, corresponding to a change of structural element. Using this strategy, we obtained the AUC of 0.80, better than all the other training models (Figure 16).

Finally, we retrieved data from lncSLdb and we asked whether was possible to predict a Nucleus/cytoplasm label. Thus, we applied the same procedure as before splitting the dataset in training set, test set and validation and with the same parameters we trained a multiclass model.

In this case the performance is lower and the AUC is about 0.70 (Figure 17)

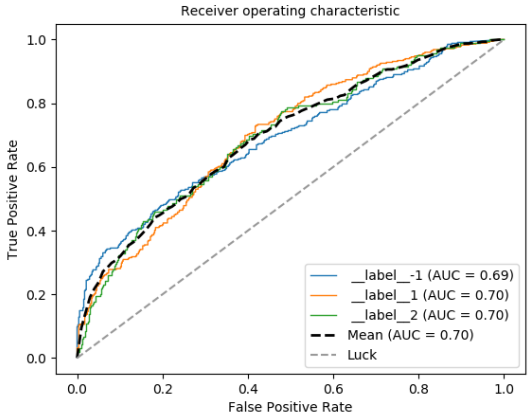


Figure 17: multiclass model. Label 1 for nuclear transcripts, label -1 for cytoplasmic transcripts and 2 for transcripts annotated with double localization (Nucleus/Cytoplasm)

4.3. Prediction of protein - RNA interactions

The success of the learning models for the inference of subcellular localization encouraged us to tackle the more complex problem of the protein-RNA interaction inference. Multiple issues must be solved in order to obtain an effective learning and good inference accuracy. The first issue is related to the input dataset construction. In a two-labels classification, each data point is assigned one of two possible labels, and the algorithm learns to discriminate between the two. In this case, one class is represented by interacting protein-lncRNA pairs extracted from the currently available databases (such as RAIN), but it is not trivial to define the other label and to collect examples belonging to this second class. If one class is represented by interacting pairs, it is natural to take as the other class a set of non-interacting pairs. Yet, it is not possible to extract such a negative dataset from existing databases, since only interactions are experimentally detectable. The adopted strategy was to shuffle the protein-lncRNA pairs in the RAIN database. By randomly pairing proteins and RNAs, one can generate a set of pairs that were never experimentally identified as interaction partners. We cannot be sure that a randomly generated pair is truly a negative, meaning that the randomly paired protein and RNA might be interacting partners that were not yet detected in any experiment. Still, the presence of such false negatives in the negative dataset could only impair the training, not favour it, therefore the resulting inference accuracy can be

considered at worst an underestimation of the one that could have been obtained having a perfect negative set.

Another problem is that each data point is now composed by two molecules of radically different nature, a protein and an RNA, both that must be encoded properly into a unique feature vector.

For both datasets (interacting pairs and negative controls), RNA secondary structure was predicted by RNAfold, the dot-bracket output was converted into the qBEAR string of structural context symbols.

We again retrieved protein-lncRNAs interaction from RAIN database (see Materials and Methods). In total, we collected 12,104 entries (6,052 for the positive and 6,052 for the negative dataset). As before, we filtered out transcripts with a length higher than 10,000 nt. We then applied a sliding window to both molecules to generate *k*-mers, testing values of *k* from 2 to 8, and considering the protein primary sequence and the lncRNAs secondary structure described in qBEAR alphabet. Result are shown in Table 10, from which it can be observed that AUC reaches a plateau for $k>3$. This prediction accuracy, while high, is inferior to that of other methods for protein-lncRNA interaction prediction, but these methods often rely on many sources of additional information, while our method employs only sequence and secondary structure.

Table 10: FastText models and AUC

Sliding window length	**AUC**
2	0.63
3	0.73
4	0.77
5	0.77
6	0.78
7	0.78
8	0.78

We compared these results with those that can be obtained using different training models, encoding the input data using again a sliding window approach, but limited to di- and tri-mer combinations; we stopped to 3mer because of the computational cost.

The results of the prediction accuracy, in terms of AUC, for these models are listed in Table 11:

Table 11: comparison of predictive score of 6 models using 2 and 3 mer approach

Learning models	kmers AUC	
	2	3
KNN	0.7	0.5
DTC	0.53	0.55
RF	0.75	0.78
EXTRATC	0.72	0.8
ABoosTC	0.53	0.85
GBoosTC	0.63	0.7

In this case, also EXTRATC and Random Forest models give good results, however the drawbacks in using a *k*-mer approach is that the number of variables depends on the combinations of characters. In our case, considering the qBEAR and the protein alphabets means that our dataset has 12,016 rows and 8485 columns. This causes a very heavy computational cost and a high intensity of training (1 day). With the FastText model and a window size of 6 bases we obtained an AUC of 0.78 (Figure 18). The time of training is considerably lower (40 min). We increased the size of sliding window, however after six bases the results do not change.

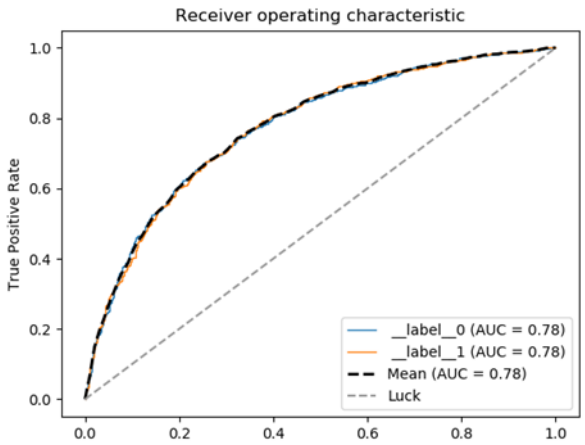


Figure 18: ROC Curve window slide of length 6

5. Conclusion and perspectives

The aim of the work was to build a heterogeneous network in order to investigate the complex class of lncRNAs. The association with proteins are of paramount importance to understand development and differentiation. Despite of the research effort, difficulties associated with the experimental determination of protein-RNA complexity and with the non-homogenous information, led to an urgent need for tools to visualize and highlight the main features between groups of interacting molecules. We started from a general view of protein-ncRNAs interaction and by a systematic approach we improved the network view by adding useful features expression values and subcellular localization. In particular, we decided to use a multidisciplinary approach based on the combination of network analysis, machine learning and Natural Language Approach treating the biological sequences as strings of characters.

In summary,

- we suggested a new way to depict the biological molecules, overcoming the drawback of the varying length that particularly affects the lncRNA class.
- we demonstrated the advantage in using the BEAR and qBEAR alphabet to encode the secondary structure. In particular qBEAR is simpler, shorter and easily understandable, hence suitable for a clear visualization.

Different tools are designed to accurately distinguish interactive and not interactive biomolecules, as well as bound from unbound sites, but a tool for lncRNA subcellular localization inference is lacking. We proposed a subcellular localization predictor that is considerably fast (20 minutes for training), easy to use, and can be applied to every set of RNA sequences. The method was trained on subcellular localization data from RNALocate, lncATLAS and lncSLdb, as well applied on protein-RNA interactions (from the RAIN database).

In order to train the model, a dot and bracket structure is needed. The output was then translated in qBEAR alphabet by a python script.

In this step, the slowest part is the secondary structure prediction. Hence, it could be useful to try other methods in order to obtain accurate structure for the input RNA sequences, as well as to use high-throughput base-pairing depiction data from techniques such as PARS or SHAPE.

Furthermore, the method defines a combined sequence/structure procedure that in turn allows to tackle biological problems where both are relevant. After training, the results can be visualized as an intuitive graph (examples are showed in Figure 21 and Figure 22) and the most frequent words mapped into original dataset to be analysed in a further way (e.g. looking for the position in the sequence, or search differences in frequency between the negative and positive dataframe).

This study can be in principle be applied to every set of RNA sequences without limit in length. As future perspectives, the method will be used to build a more complex lncRNA-protein network that also include our localization and interaction predictions. It may be also applied to CLIP datasets to discover binding sites. In addition, there are several advantages in using an alphabet to encode the secondary structure. The most evident is in analysing and interpreting the results, since the fastText model is not a black box like many other training models. For example, it is easy to map the most frequent words (see Appendix B for a list of such words) to a sequence to study their position as well retrieve information about the context, once a strategy that gives a reliable score is found. Words with a positive and negative impact are provided by the software (Figure 19 and Figure 20), and could be in principle associated, individually or as groups forming sentences, to localization or interaction signals within the sequence.

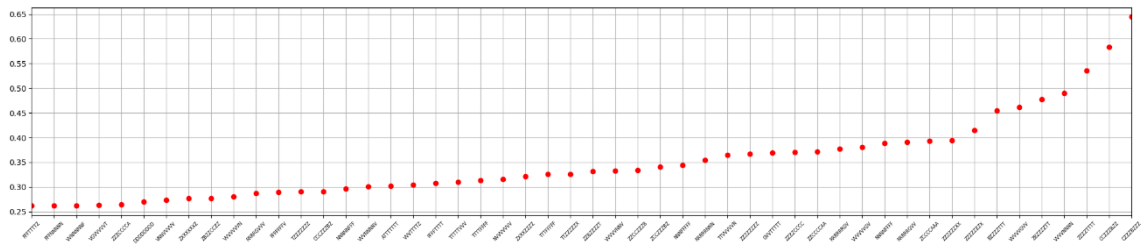


Figure 19: elements with a positive score

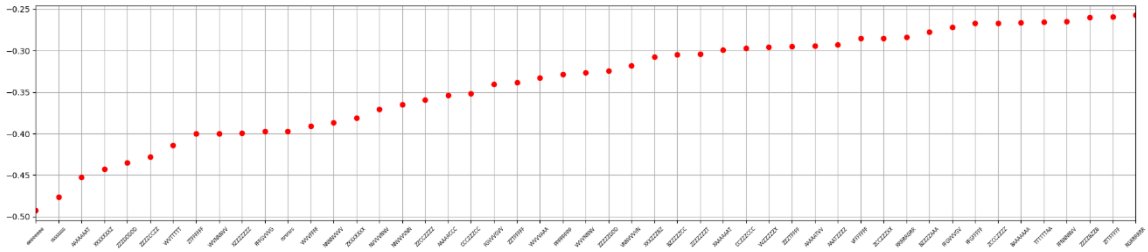


Figure 20 : elements with a negative score

We also started to design an approach to compare different classes of lncRNAs (e.g. miRNA and lncRNAs) in order to find similarities in sequences. Again, in this context it may be useful a network visualization of significant words along the RNA sequence. These approaches can be used for pairwise comparisons, to highlight shared words and their location between two RNAs (Figure 19) or to globally compare large sets of molecules and their relationships (Figure 20). For what concerns the availability all procedure may be implementable as a docker hub and scripts in a GitHub page

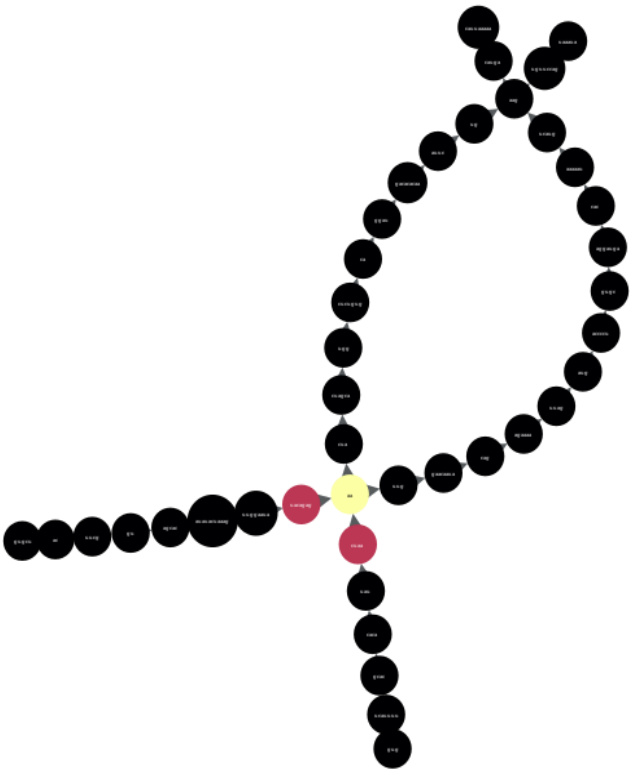


Figure 21: Example of two sequences, betweenness value is highlighted

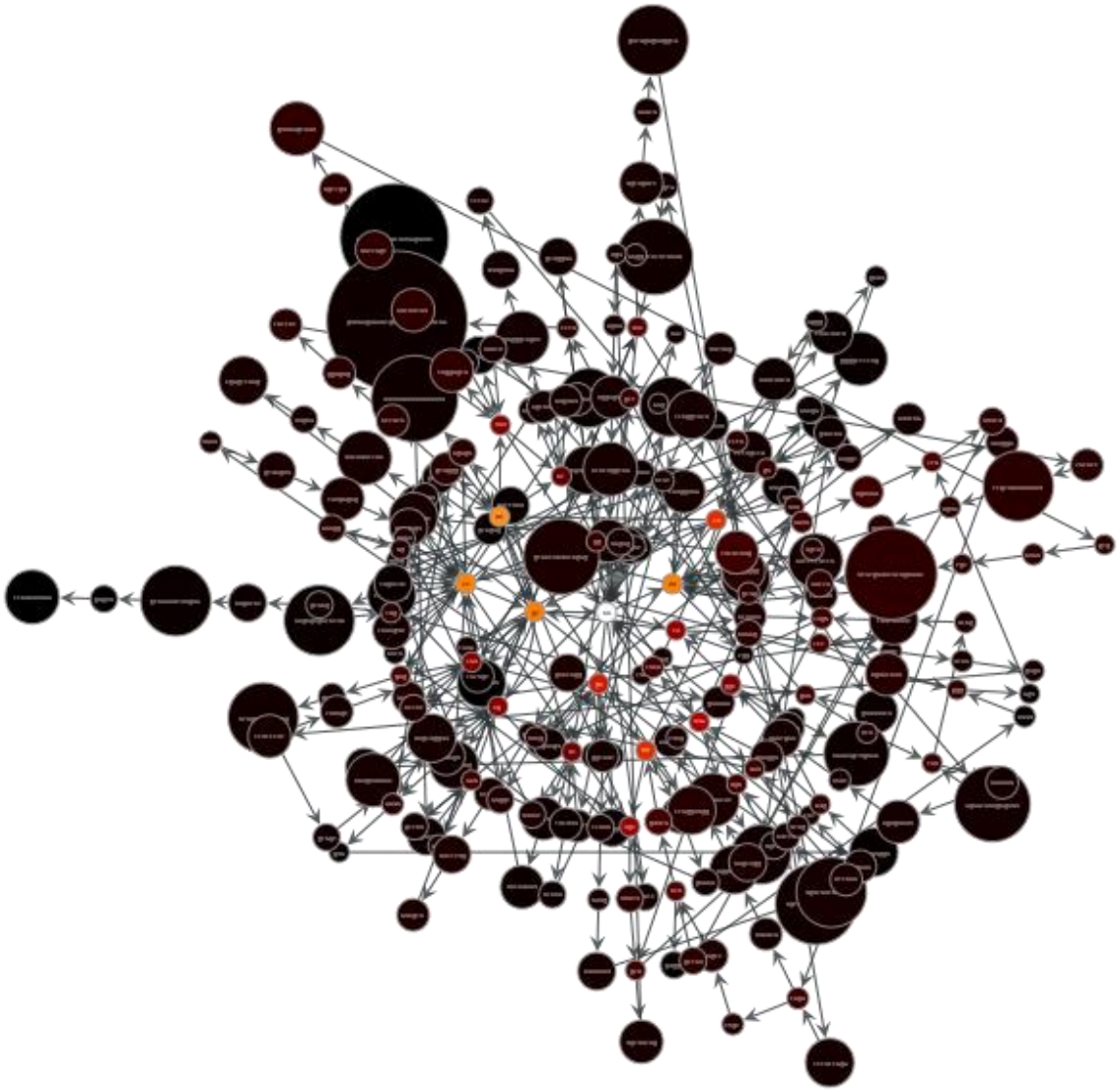


Figure 22: short-lncRNAs similarity

References

1. Guo, X.; Gao, L.; Wang, Y.; Chiu, D. K. Y.; Wang, T.; Deng, Y. Advances in long noncoding RNAs: Identification, structure prediction and function annotation. *Brief. Funct. Genomics* **2016**, *15*, 38–46, doi:10.1093/bfgp/elv022.
2. Fang, S.; Zhang, L.; Guo, J.; Niu, Y.; Wu, Y.; Li, H.; Zhao, L.; Li, X.; Teng, X.; Sun, X.; Sun, L.; Zhang, M. Q.; Chen, R.; Zhao, Y. NONCODEV5: A comprehensive annotation database for long non-coding RNAs. *Nucleic Acids Res.* **2018**, *46*, D308–D314, doi:10.1093/nar/gkx1107.
3. Chen, L. L. Linking Long Noncoding RNA Localization and Function. *Trends Biochem. Sci.* **2016**, *41*, 761–772, doi:10.1016/j.tibs.2016.07.003.
4. Long, Y.; Wang, X.; Youmans, D. T.; Cech, T. R. How do lncRNAs regulate transcription? *Sci. Adv.* **2017**, *3*, doi:10.1126/sciadv.aao2110.
5. Junge, A.; Refsgaard, J. C.; Garde, C.; Pan, X.; Santos, A.; Alkan, F.; Anthon, C.; Von Mering, C.; Workman, C. T.; Jensen, L. J.; Gorodkin, J. RAIN: RNA-protein Association and Interaction Networks. *Database* **2017**, *2017*, 1–9, doi:10.1093/database/baw167.
6. Quek, X. C.; Thomson, D. W.; Maag, J. L. V; Bartonicek, N.; Signal, B.; Clark, M. B.; Gloss, B. S.; Dinger, M. E. lncRNAdb v2.0: Expanding the reference database for functional long noncoding RNAs. *Nucleic Acids Res.* **2015**, *43*, D168–D173, doi:10.1093/nar/gku988.
7. Zhang, T.; Tan, P.; Wang, L.; Jin, N.; Li, Y.; Zhang, L.; Yang, H.; Hu, Z.; Zhang, L.; Hu, C.; Li, C.; Qian, K.; Zhang, C.; Huang, Y.; Li, K.; Lin, H.; Wang, D. RNALocate: A resource for RNA subcellular localizations. *Nucleic Acids Res.* **2017**, *45*, D135–D138, doi:10.1093/nar/gkw728.
8. Volders, P. J.; Verheggen, K.; Menschaert, G.; Vandepoele, K.; Martens, L.; Vandesompele, J.; Mestdagh, P. An update on LNCipedia: A database for annotated human lncRNA sequences. *Nucleic Acids Res.* **2015**, *43*, D174–D180, doi:10.1093/nar/gku1060.
9. Molecular, E.; Genome, W. T.; Road, O.; Trust, W.; Campus, G.; Antonio, S.; Laboratories, S. N.; Science, F.; Lansing, E.; Systems, C.; Centre, B.; Building, S.; Resource, A. I.; Bioinformatics, P.; City, R.; Science, D.; Group, B.; Unit, G. RNAcentral: a comprehensive database of non-coding RNA sequences. *Nucleic Acids Res.* **2017**, *45*, D128–D134, doi:10.1093/nar/gkw1008.
10. Yates, A.; Akanni, W.; Amode, M. R.; Barrell, D.; Billis, K.; Carvalho-Silva, D.; Cummins, C.; Clapham, P.; Fitzgerald, S.; Gil, L.; Girón, C. G.; Gordon, L.; Hourlier, T.; Hunt, S. E.; Janacek, S. H.; Johnson, N.; Juettemann, T.; Keenan, S.; Lavidas, I.; Martin, F. J.; Maurel, T.; McLaren, W.; Murphy, D. N.; Nag, R.; Nuhn, M.; Parker, A.; Patricio, M.; Pignatelli, M.; Rahtz, M.; Riat, H. S.; Sheppard, D.; Taylor, K.; Thormann, A.; Vullo, A.; Wilder, S. P.; Zadissa, A.; Birney, E.; Harrow, J.; Muffato, M.; Perry, E.; Ruffier, M.; Spudich, G.; Trevanion, S. J.; Cunningham, F.; Aken, B. L.; Zerbino, D. R.; Flicek, P. Ensembl 2016. *Nucleic Acids Res.* **2016**, *44*, D710–D716, doi:10.1093/nar/gkv1157.
11. Hao, Y.; Wu, W.; Li, H.; Yuan, J.; Luo, J.; Zhao, Y.; Chen, R. NPInter v3.0: An upgraded database of noncoding RNA-associated interactions. *Database* **2016**, *2016*, 1–9, doi:10.1093/database/baw057.
12. Heller, D.; Krestel, R.; Ohler, U.; Vingron, M.; Marsico, A. SSHMM: Extracting intuitive sequence-structure motifs from high-Throughput RNA-binding protein data. *Nucleic Acids Res.* **2017**, *45*, 11004–11018, doi:10.1093/nar/gkx756.
13. Maticzka, D.; Lange, S. J.; Costa, F.; Backofen, R. GraphProt: Modeling binding

preferences of RNA-binding proteins. *Genome Biol.* **2014**, *15*, 1–18, doi:10.1186/gb-2014-15-1-r17.
14. Kazan, H.; Ray, D.; Chan, E. T.; Hughes, T. R.; Morris, Q. RNAcontext: A new method for learning the sequence and structure binding preferences of RNA-binding proteins. *PLoS Comput. Biol.* **2010**, *6*, 28, doi:10.1371/journal.pcbi.1000832.
15. Guttman, M.; Rinn, J. L. Modular regulatory principles of large non-coding RNAs. *Nature* **2012**, *482*, 339–346, doi:10.1038/nature10887.
16. Rao, M. R. S. *Long Non Coding RNA Biology*; Advances in Experimental Medicine and Biology; Springer Singapore, 2017; ISBN 9789811052033.
17. Aw, J. G. A.; Shen, Y.; Wilm, A.; Sun, M.; Lim, X. N.; Boon, K. L.; Tapsin, S.; Chan, Y. S.; Tan, C. P.; Sim, A. Y. L.; Zhang, T.; Susanto, T. T.; Fu, Z.; Nagarajan, N.; Wan, Y. In Vivo Mapping of Eukaryotic RNA Interactomes Reveals Principles of Higher-Order Organization and Regulation. *Mol. Cell* **2016**, *62*, 603–617, doi:10.1016/j.molcel.2016.04.028.
18. McHugh, C. A.; Russell, P.; Guttman, M. Methods for comprehensive experimental identification of RNA-protein interactions. *Genome Biol.* **2014**, *15*, 1–10, doi:10.1186/gb4152.
19. Kudla, G.; Granneman, S.; Hahn, D.; Beggs, J. D.; Tollervey, D. Cross-linking, ligation, and sequencing of hybrids reveals RNA-RNA interactions in yeast. *Proc. Natl. Acad. Sci.* **2011**, *108*, 10010–10015, doi:10.1073/pnas.1017386108.
20. Ferrè, F.; Colantoni, A.; Helmer-Citterich, M. Revealing protein-lncRNA interaction. *Brief. Bioinform.* **2016**, *17*, 106–116, doi:10.1093/bib/bbv031.
21. Keene, J. D.; Komisarow, J. M.; Friedersdorf, M. B. RIP-Chip: the isolation and identification of mRNAs, microRNAs and protein components of ribonucleoprotein complexes from cell extracts. *Nat. Protoc.* **2006**, *1*, 302.
22. Zhao, J.; Ohsumi, T. K.; Kung, J. T.; Ogawa, Y.; Daniel, J.; Sarma, K.; Song, J. J.; Kingston, R. E.; Borowsky, M.; Lee, J. T. NIH Public Access. **2011**, *40*, 939–953, doi:10.1016/j.molcel.2010.12.011.Genome-wide.
23. Van Nostrand, E. L.; Pratt, G. A.; Shishkin, A. A.; Gelboin-Burkhart, C.; Fang, M. Y.; Sundararaman, B.; Blue, S. M.; Nguyen, T. B.; Surka, C.; Elkins, K.; Stanton, R.; Rigo, F.; Guttman, M.; Yeo, G. W. Robust transcriptome-wide discovery of RNA-binding protein binding sites with enhanced CLIP (eCLIP). *Nat. Methods* **2016**, *13*, 508–514, doi:10.1038/nmeth.3810.
24. Zhao, Y.; Granas, D.; Stormo, G. D. Inferring binding energies from selected binding sites. *PLoS Comput. Biol.* **2009**, *5*, doi:10.1371/journal.pcbi.1000590.
25. Campbell, Z. T.; Bhimsaria, D.; Valley, C. T.; Rodriguez-martinez, J. a; Menichelli, E.; Williamson, J. R.; Ansari, A. Z.; Wickens, M. NIH Public Access. *Cell* **2012**, *1*, 570–581, doi:10.1016/j.celrep.2012.04.003.Licensing.
26. Szeto, K.; Latulippe, D. R.; Ozer, A.; Pagano, J. M.; White, B. S.; Shalloway, D.; Lis, J. T.; Craighead, H. G. RAPID-SELEX for RNA aptamers. *PLoS One* **2013**, *8*, doi:10.1371/journal.pone.0082667.
27. Ray, D.; Kazan, H.; Chan, E. T.; Castillo, L. P.; Chaudhry, S.; Talukder, S.; Blencowe, B. J.; Morris, Q.; Hughes, T. R. Rapid and systematic analysis of the RNA recognition specificities of RNA-binding proteins. *Nat. Biotechnol.* **2009**, *27*, 667–670, doi:10.1038/nbt.1550.
28. Lambert, N. J.; Robertson, A. D.; Burge, C. B. RNA Bind-n-Seq: Measuring the Binding Affinity Landscape of RNA Binding Proteins. *Methods Enzymol.* **2015**, *558*, 465–493, doi:10.1016/bs.mie.2015.02.007.
29. Kozomara, A.; Griffiths-Jones, S. MiRBase: Annotating high confidence microRNAs using deep sequencing data. *Nucleic Acids Res.* **2014**, *42*, 68–73,

doi:10.1093/nar/gkt1181.
30. Chen, L. L.; Carmichael, G. G. Altered Nuclear Retention of mRNAs Containing Inverted Repeats in Human Embryonic Stem Cells: Functional Role of a Nuclear Noncoding RNA. *Mol. Cell* **2009**, doi:10.1016/j.molcel.2009.06.027.
31. Ip, J. Y.; Nakagawa, S. Long non-coding RNAs in nuclear bodies. *Dev. Growth Differ.* **2012**, *54*, 44–54, doi:10.1111/j.1440-169X.2011.01303.x.
32. Brown, C. J.; Hendrich, B. D.; Rupert, J. L.; Lafrenière, R. G.; Xing, Y.; Lawrence, J.; Willard, H. F. The human XIST gene: Analysis of a 17 kb inactive X-specific RNA that contains conserved repeats and is highly localized within the nucleus. *Cell* **1992**, *71*, 527–542, doi:https://doi.org/10.1016/0092-8674(92)90520-M.
33. Castelnuovo, M.; Rahman, S.; Guffanti, E.; Infantino, V.; Stutz, F.; Zenklusen, D. Bimodal expression of PHO84 is modulated by early termination of antisense transcription. *Nat. Struct. Mol. Biol.* **2013**, *20*, 851–858, doi:10.1038/nsmb.2598.
34. Lee, J. H.; Daugharthy, E. R.; Scheiman, J.; Kalhor, R.; Ferrante, T. C.; Terry, R.; Turczyk, B. M.; Yang, J. L.; Lee, H. S.; Aach, J.; Zhang, K.; Church, G. M. Fluorescent in situ sequencing (FISSEQ) of RNA for gene expression profiling in intact cells and tissues. *Nat. Protoc.* **2015**, *10*, 442–458, doi:10.1038/nprot.2014.191.
35. Lee, J. H. Quantitative approaches for investigating the spatial context of gene expression. *Wiley Interdiscip. Rev. Syst. Biol. Med.* **2017**, *9*, 17–19, doi:10.1002/wsbm.1369.
36. Mas-Ponte, D.; Carlevaro-Fita, J.; Palumbo, E.; Hermoso Pulido, T.; Guigo, R.; Johnson, R. LncATLAS database for subcellular localisation of long noncoding RNAs. *Rna* **2017**, *23*, 1080–1087, doi:10.1101/116335.
37. Ulitsky, I.; Bartel, D. P. Leading Edge Review lincRNAs: Genomics, Evolution, and Mechanisms. **2013**, doi:10.1016/j.cell.2013.06.020.
38. Engreitz, J. M.; Ollikainen, N.; Guttman, M. Long non-coding RNAs: Spatial amplifiers that control nuclear structure and gene expression. *Nat. Rev. Mol. Cell Biol.* **2016**, *17*, 756–770, doi:10.1038/nrm.2016.126.
39. Underwood, J. G.; Uzilov, A. V.; Katzman, S.; Onodera, C. S.; Mainzer, J. E.; Mathews, D. H.; Lowe, T. M.; Salama, S. R.; Haussler, D. FragSeq: Transcriptome-wide RNA structure probing using high-throughput sequencing. *Nat. Methods* **2010**, *7*, 995–1001, doi:10.1038/nmeth.1529.
40. Wilkinson, K. A.; Merino, E. J.; Weeks, K. M. Selective 2′-hydroxyl acylation analyzed by primer extension (SHAPE): quantitative RNA structure analysis at single nucleotide resolution. *Nat. Protoc.* **2006**, *1*, 1610.
41. Lu, Z.; Gong, J.; Zhang, Q. C. RNA Detection. **2018**, *1649*, 59–84, doi:10.1007/978-1-4939-7213-5.
42. Ledda, M.; Aviran, S. PATTERNA: Transcriptome-wide search for functional RNA elements via structural data signatures. *Genome Biol.* **2018**, *19*, 1–18, doi:10.1186/s13059-018-1399-z.
43. Suresh, V.; Liu, L.; Adjeroh, D.; Zhou, X. RPI-Pred: Predicting ncRNA-protein interaction using sequence and structural information. *Nucleic Acids Res.* **2015**, *43*, 1370–1379, doi:10.1093/nar/gkv020.
44. Lorenz, R.; Bernhart, S. H.; zu Siederdissen, C.; Tafer, H.; Flamm, C.; Stadler, P. F.; Hofacker, I. L. ViennaRNA Package 2.0. *Algorithms Mol. Biol.* **2011**, *6*, 26, doi:10.1186/1748-7188-6-26.
45. Reuter, J. S.; Mathews, D. H. RNAstructure: software for RNA secondary structure prediction and analysis. *BMC Bioinformatics* **2010**, *11*.
46. Steffen, P.; Voß, B.; Rehmsmeier, M.; Reeder, J.; Giegerich, R. RNAshapes: an

integrated RNA analysis package based on abstract shapes. *Bioinformatics* **2006**, *22*, 500–503.
47. Adjeroh, D.; Allaga, M.; Tan, J.; Lin, J.; Jiang, Y.; Abbasi, A.; Zhou, X. Feature-based and string-based models for predicting RNA-protein interaction. *Molecules* **2018**, *23*, 1–17, doi:10.3390/molecules23030697.
48. No Title Available online: https://pypi.org/project/forgi.
49. Mattei, E.; Ausiello, G.; Ferrè, F.; Helmer-Citterich, M. A novel approach to represent and compare RNA secondary structures. *Nucleic Acids Res.* **2014**, *42*, 6146–6157, doi:10.1093/nar/gku283.
50. Gardner, P.; Daub, J.; Tate, J.; Nawrocki, E.; Kolbe, D.; Lindgreen, S.; Wilkinson, A.; Finn, R.; Griffiths-Jones, S.; Eddy, S. Rfam: updates to the RNA families database. *Nucleic Acids Res.* **2009**, *37*.
51. Kirk, J. M.; Kim, S. O.; Inoue, K.; Smola, M. J.; Lee, D. M.; Schertzer, M. D.; Wooten, J. S.; Baker, A. R.; Sprague, D.; Collins, D. W.; Horning, C. R.; Wang, S.; Chen, Q.; Weeks, K. M.; Mucha, P. J.; Calabrese, J. M. Functional classification of long non-coding RNAs by k-mer content. *Nat. Genet.* **2018**, *50*, 1474–1482, doi:10.1038/s41588-018-0207-8.
52. Pancaldi, V.; Bähler, J. In silico characterization and prediction of global protein-mRNA interactions in yeast. *Nucleic Acids Res.* **2011**, *39*, 5826–5836, doi:10.1093/nar/gkr160.
53. Agostini, F.; Zanzoni, A.; Klus, P.; Marchese, D.; Cirillo, D.; Tartaglia, G. G. CatRAPID omics: A web server for large-scale prediction of protein-RNA interactions. *Bioinformatics* **2013**, *29*, 2928–2930, doi:10.1093/bioinformatics/btt495.
54. Lu, Q.; Ren, S.; Lu, M.; Zhang, Y.; Zhu, D.; Zhang, X.; Li, T. Computational prediction of associations between long non-coding RNAs and proteins. *BMC Genomics* **2013**, *14*, 1, doi:10.1186/1471-2164-14-651.
55. Muppirala, U. K.; Honavar, V. G.; Dobbs, D. Predicting RNA-Protein Interactions Using Only Sequence Information. *BMC Bioinformatics* **2011**, *12*, 489, doi:10.1186/1471-2105-12-489.
56. Alipanahi, B.; Delong, A.; Weirauch, M. T.; Frey, B. J. Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat. Biotechnol.* **2015**, *33*, 831–838, doi:10.1038/nbt.3300.
57. Pan, X.; Fan, Y. X.; Yan, J.; Shen, H. Bin IPMiner: Hidden ncRNA-protein interaction sequential pattern mining with stacked autoencoder for accurate computational prediction. *BMC Genomics* **2016**, *17*, 1–14, doi:10.1186/s12864-016-2931-8.
58. Singh-Blom, U. M.; Natarajan, N.; Tewari, A.; Woods, J. O.; Dhillon, I. S.; Marcotte, E. M. Prediction and Validation of Gene-Disease Associations Using Methods Inspired by Social Network Analyses. *PLoS One* **2013**, *8*, doi:10.1371/journal.pone.0058977.
59. Li, A.; Ge, M.; Zhang, Y.; Peng, C.; Wang, M. Predicting Long Noncoding RNA and Protein Interactions Using Heterogeneous Network Model. *Biomed Res Int* **2015**, *2015*, 671950, doi:10.1155/2015/671950.
60. Xiao, Y.; Zhang, J.; Deng, L. Prediction of lncRNA-protein interactions using HeteSim scores based on heterogeneous networks. *Sci. Rep.* **2017**, *7*, 1–12, doi:10.1038/s41598-017-03986-1.
61. Barik, A.; Mishra, A.; Bahadur, R. P. PRince: A web server for structural and physicochemical analysis of Protein-RNA interface. *Nucleic Acids Res.* **2012**, *40*, 440–444, doi:10.1093/nar/gks535.

62. Nussinov, R.; Jacobson, A. B. Fast algorithm for predicting the secondary structure of single stranded RNA. *Proc Natl Acad Sci USA* **1980**, *77*.
63. Shapiro, B. A.; Zhang, K. Comparing multiple RNA secondary structures using tree comparisons. *Bioinformatics* **1990**, *6*, 309–318, doi:10.1093/bioinformatics/6.4.309.
64. Benedetti, G.; Morosetti, S. A graph-topological approach to recognition of pattern and similarity in RNA secondary structures. *Biophys. Chem.* **1996**, *59*, 179–184, doi:10.1016/0301-4622(95)00119-0.
65. Fera, D.; Kim, N.; Shiffeldrim, N.; Zorn, J.; Laserson, U.; Gan, H. H.; Schlick, T. RAG: RNA-As-Graphs web resource. *BMC Bioinformatics* **2004**, *5*, 1–9, doi:10.1186/1471-2105-5-88.
66. von Mering, C.; Huynen, M.; Jaeggi, D.; Schmidt, S.; Bork, P.; Snel, B. STRING: A database of predicted functional associations between proteins. *Nucleic Acids Res.* **2003**, *31*, 258–261, doi:10.1093/nar/gkg034.
67. Hmisc.
68. DataExplorer Available online: https://cran.r-project.org/web/packages/DataExplorer.
69. godb Available online: 10.18129/B9.bioc.GO.db.
70. Carlson, M.; Aboyoun, P.; Pages, H.; Falcon, S.; Morgan, M. Making and Utilizing TxDb Objects. *Bioconductor* **2017**, 1–13.
71. biomaRt Available online: 10.18129/B9.bioc.biomaRt.
72. Pandas.
73. scikit Available online: scikit-learn.org/.
74. fastText Available online: https://fasttext.cc/.
75. Hsu, S. Da; Lin, F. M.; Wu, W. Y.; Liang, C.; Huang, W. C.; Chan, W. L.; Tsai, W. T.; Chen, G. Z.; Lee, C. J.; Chiu, C. M.; Chien, C. H.; Wu, M. C.; Huang, C. Y.; Tsou, A. P.; Huang, H. Da MiRTarBase: A database curates experimentally validated microRNA-target interactions. *Nucleic Acids Res.* **2011**, *39*, 163–169, doi:10.1093/nar/gkq1107.
76. Li, J. H.; Liu, S.; Zhou, H.; Qu, L. H.; Yang, J. H. StarBase v2.0: Decoding miRNA-ceRNA, miRNA-ncRNA and protein-RNA interaction networks from large-scale CLIP-Seq data. *Nucleic Acids Res.* **2014**, *42*, 92–97, doi:10.1093/nar/gkt1248.
77. miRanda Available online: http://www.mirtoolsgallery.org/miRToolsGallery/node/1055.
78. PITA Available online: https://genie.weizmann.ac.il/pubs/mir07/mir07_data.html.
79. TargetScan.
80. miRDB.
81. starmirDB.

Appendix A

Table 12 Results for the proposed fastext model on RNALocate dataset

	Primary sequences	Bear encoded structure	qBear encoded structure
Slw2	0.78	0.68	0.63
Slw3	0.79	0.70	0.63
Slw4	0.80	0.74	0.66
splitbydifferentchar		0.77	0.78
splitbyonecharacter		0.70	0.72
splitbypattern		0.79	0.80

Table 13 Results for the proposed fastext model on lncATLAS

	Primary sequences	Bear encoded structure	qBear encoded structure
Slw2	0.56	0.5	0.6
Slw3	0.53	0.5	0.5
Slw4	0.57	0.5	0.5
splitbydifferentchar		0.6	0.6
splitbyonecharacter		0.5	0.5
splibypattern		0.6	0.73

Appendix B

Table 14: most frequent words in localization prediction

N	word	N2	word3	N4	word5	N6	word7
1	C	25	CCU	50	GGU	75	AAAA
2	U	26	GGA	51	UGU	76	UCCU
3	G	27	GCU	52	CAA	77	GCUG
4	GC	28	UCU	53	CAU	78	GGCC
5	CA	29	UCC	54	GUC	79	CCUG
6	AA	30	CUC	55	AAG	80	CAGG
7	CC	31	AGA	56	ACU	81	UAC
8	GA	32	UGG	57	GAC	82	GCAG
9	CU	33	GAG	58	ACC	83	GGGC
10	GG	34	CCC	59	AAU	84	GCCU
11	AG	35	GCA	60	AGU	85	CUCC
12	UC	36	UCA	61	GUU	86	CCAG
13	UG	37	AGG	62	AUU	87	UUCU
14	UU	38	ACA	63	AAC	88	CCCA
15	AC	39	GAA	64	UUG	89	GCCC
16	GU	40	GGG	65	AUG	90	GGCU
17	AU	41	UGC	66	AUA	91	GAGG
18	UA	42	AGC	67	AUC	92	GGGA
19	GCC	43	CG	68	GAU	93	CUGG
20	GGC	44	CUU	69	UAA	94	UGGG
21	CAG	45	UUU	70	UUA	95	GAAA
22	AAA	46	GUG	71	GGAG	96	UCUG
23	CCA	47	CAC	72	CUA	97	CUGC
24	CUG	48	UUC	73	UAU	98	CAGA
25	CCU	49	UGA	74	GUA		

Table 15 : most 50 positive words in protein-RNA prediction. Upper case for lncRNA secondary structure, qBEAR alphabet lower case for protein sequence

N	word	N2	word3
1	eeeeeeee	26	ZZZZZDDD
2	ssssssss	27	VNNVVVVN
3	AAAAAAAT	28	XXXZZZBZ
4	XXXXXXXZ	29	BZZZZZCC
5	ZZZDDDDD	30	ZZZZZZZT
6	ZZZZCCZZ	31	XAAAAAAT
7	VVVTTTTT	32	CCZZZCCC
8	ZTFFFFFF	33	VVZZZZZX
9	VVVNNNVV	34	ZZZTFFFF
10	XZZZZZZZ	35	AAAAATVV
11	FFFGVVVG	36	AAATZZZZ
12	rsrsrsrs	37	VFFFFFFF
13	VVVVFFFF	38	ZCCZZZXX
14	NNNNVVVV	39	RRRRRGRR
15	ZXXXXXXX	40	BZZZZAAA
16	NVVVVNNV	41	FFGVVVGV
17	NNVVVVNN	42	FFGFFFFF
18	ZZCCZZZZ	43	ZCCCZZZZ
19	AAAAACCC	44	BAAAAAAA
20	CCCZZZCC	45	TTTTTTAA
21	FGVVVGVV	46	FFNNNNVV
22	ZZTFFFFF	47	ZZZZBZZB
23	VVVVVAAA	48	ZTTFFFFF
24	pppppppp	49	RGRRRRRR
25	VVVVNNNV		